Object Modeling with Conic Splines

by

Mohammed Aiyaz Hussain

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

December, 1997

OBJECT MODELING WITH CONIC SPLINES

BY

MOHAMMED AIYAZ HUSSAIN

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

DECEMBER 1997

UMI Number: 1388275

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN 31261, SAUDI ARABIA

COLLEGE OF GRADUATE STUDIES

This thesis, written by MOHAMMED AIYAZ HUSSAIN under the direction of his Thesis Advisor and approved by his Thesis Committee, has been presented to and accepted by the Dean of the College of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE** in **COMPUTER SCIENCE.**

THESIS COMMITTEE

Dr. Muhammed Sarfraz (Thesis Advisor)

Dr. Jarallah AlGhamdi (Member)

Dr. Muhammed AlMulhem (Member)

Dr. Mostafa Abd – El – Barr (Member)

Dec 10, 97

Department Chairman

Dean, College of Graduate Studies

13-12-97

Date

COLLEGE OF GRADUATE STUDIES

كلية الدراسات العليا

Dedicated to

Mummy, Baba, Shaaz

and

Somebody very close to my Heart

ACKNOWLEDGEMENTS

In the name of Allah, Most Gracious, Most Merciful.

Read in the name of thy Lord and Cherisher, Who created. Created man from a leech-like clot. Read and thy Lord is Most Bountiful. He who taught the use of pen. Taught man that which he knew not. Nay, but man doth trangress all bounds, in that he looketh upon himself as self-sufficient. Verily, to they Lord is the return of all. **(The Holy Quran, Surah 96)**

All Praise is due to ALLAH to whom belongs the dominion of the Heavens and the Earth. Peace and mercy be upon His Prophet. I thank Him for giving me the knowledge and patience to carry out this work.

Acknowledgement is due to King Fahd University of Petroleum and Minerals for support of this research.

First and foremost, I would like to express my humble gratitudes to my parents, without whose blessings, support, encouragement and motivation, I wouldn't have been what I am, and really, I mean it. I thank ALLAH (S.W.T) for blessing me with such a caring and affectionate family. I would like to take this oppurtunity to express my acknowledgement to my Mummy, Baba and brother for the strength and patience they had shown by letting me pursue my higher studies here at KFUPM, far away from them.

I would like to offer my indebtness and sincere appreciation to my thesis advisor Dr. Muhammed Sarfraz and no word of thanks would be sufficient for his guidance. He was always available to me and always ready to help me. Special thanks are due to my thesis committee members Dr. Jaralla Al-Ghamdi, Dr. M. Al-Mulhem and Dr. Mostafa Barr for all their cooperation.

To just express thanks to my Aunts, Uncles and cousins, who stay in Dhahran, would be very bad on my part. Infact, they made me realize that there are people other than my family members who can shower so much love and affection on me. Thanks are due to my friends Ghouse, Irfan, Jaleel and all others with whom I have spent some memorable moments. Thanks are also due to the faculty and staff of the ICS Dept. with particular mention to Mr. Shahid Ali, for providing an amicable work atmosphere.

And above all, I would like to give a special vote of thanks to a person who is very special to me and who has been a constant source of inspiration to me.

Contents

List of Figures

THESIS ABSTRACT

Name: MOHAMMED AIYAZ HUSSAIN

Title: OBJECT MODELING WITH CONIC SPLINES

Degree: MASTER OF SCIENCE

Major Field: INFORMATION & COMPUTER SCIENCE

Date of Degree: DECEMBER 1997

In Computer Graphics, it is often necessary to represent a hand-drawn shape accurately. Modeling such shapes manually is both cumbersome and commercially expensive. User's concern for curves that are easily to manipulate has been a major influence on the development of free form curves. Rational parametric curves have been receiving considerable attention in the areas of geometric modeling because any parametric polynomial curve can be expressed as a rational curve and most polynomial splines and curve have rational extensions. Rational parametric curves can be used to model any object like ships, airplane, and even in the medical field for modeling heart or other parts of the body.

In object modeling rational cubic is most popular because it is the lowest degree that can define space curves and curves with points of zero curvature. The foremost objective of this work is to show how conic sections can adequately be used to represent curves and objects that were previously thought to require rational cubic splines. A single rational cubic is represented by two conics by splitting the rational cubic at its mid-point. Both the C^1 (tangent) and C^2 (curvature) continuous rational cubic and their equivalent C^1 and C^2 continuous conic respectively are investigated. It will be shown how the conic representation is advantageous over rational cubic, in terms of computational requirements and shape control.

Also, different ways of constructing the conic and different ways of improving the smoothness of conic is also investigated. Various ways of getting an offset to a conic is also studied which has a very good application in font designing. All of this work is carried on both two-dimensional curves and three-dimensional objects.

King Fahd University of Petroleum and Minerals, Dhahran.
December 1997

خلاصة الرسالة

اسم الطالب : محمد عياز حسين

عنوان الرسالة : النموذج البنائي باستخدام المنحنيات التربيعيه

التخصص : علوم الحاسب الآلي والمعلومات

تاريخ الشهادة : ٥ ديسمبر ١٩٩٧م

عند استخدام الحاسب للرسم غالباً ما يكون من الضروري ان يعمل رسم يدوي دقيق وهي عملية مكلفه وصعبه . ان اهتمام المستخدم بالمنحنيات التي يسهل التعامل معها اصبح له تأثير كبير على تطوير اشكال حرة من المنحنيات . لقد اصبح الاهتمام بالمنحنيات النسبيه ذات المتغير الواحد في مجال بناد النموذج الهندسي كبيراً لان اي داله كثيرة حدود يمكن ان تمثل كمنحى نسبي وكذلك معظم منحنيات كثيرات الحدود لها امتدادات نسبيه . ان المنحنيات النسبيه ذات المتغير الواحد يمكن ان تستخدم في عمل اي نموذج بنائي للسفن الطائرات وكذلك في المجال الطبي في عمل نموذج للقلب او اية اعضاء اخرى من الجسم البشري .

يعتبر النموذج البنائي النسبي التكعيبي اكثر شعبيه لانه اقل درجة يمكن بها تعريف المنحنيات الفضائيه والمنحنيات ذات انحناء مساوي للصفر في بعض المنقاط . ان اهمية هذا البحث تمكن في انه يعرض كيف ان القطوع المخروطيه يمكن ان تستخدم لتمثيل المنحنيات التي كان يعتقد سابقاً انها تتطلب منحنيات تكعبيه نسبيه . المنحنى التكعيبي النسبي المفرد يتم تمثيله . بآثنان من القطوع عد طريق فصل المنحنى النسبي التكعيبي من وسطه . سوف اعرض في هذا البحث كيف ان المنحنيات التربيعيه اكثر فائدة من المنحنيات التكعبيه النسبيه من حيث التحكم بالشكل واجراء العمليات الحسابيه المطلوبه .

درجة الماجستير في العلوم

جامعة الملك فهد للبترول والمعادن

الظهران - المملكة العربية السعودية

ديسمبر ١٩٩٧م

Chapter 1

INTRODUCTION

In computer graphics, it is often necessary to represent a hand-drawn shape accurately. Generating such shapes manually is both cumbersome and commercially expensive. User's concern for curves that are easy to manipulate has been a major influence on the development of free form curves. The development concentrated mainly on the definition and control of curves. The most developed is the *parametric representation*, and perhaps the polynomial form is most suitable in design. Most Computer-Aided-design (CAD) systems have a spline routine to assist the designer in this respect. The designer uses several parametric cubic segments to construct the given hand-drawn shape. *Parametric piecewise-cubic functions* are used throughout the computer graphics industry to represent curved shapes. For many applications it would be useful to be able to reliably derive this representation from a closely spaced set of points that approximate the desired curve, such as the input from a digitizing tablet or a scanner.

Rational parametric curves have been receiving considerable attention in the areas of computer graphics and geometric modeling. This is due in part to the fact that most common primitives such as conics as well as free form geometry can be represented by this single formulation. Any parametric polynomial curve can be expressed as a rational curve, and most polynomial splines and curve have rational extensions. But since a single rational function usually does not have enough freedom to represent a given curve, several rational segments are used instead. To generate a curve of satisfactory smoothness, the segments must connect with some amount of *continuity.* Thus, the use of rational curves, independent of the particular variety, creates a common problem that of connecting rational segments to form piecewise rational curves that are smooth.

Curves that are easy to control are useful in design. One of the most accepted and elegant technique used to represent input shape information has been developed by Bezier [1] and [2], which is perhaps the most popular in CADS. In CAD, the *cubic* is the most popular because it is the lowest degree that can define space curves and curves with inflections which has zero curvature.

Conic curve segments can also be used to produce a required shape. In [3] Pratt has shown that conic splines could be used to approximate some of the properties of cubic curves and also developed a means of generating conic splines using an all integer version of Pitteway's algorithm [2]. The method developed by Pratt in [3] forms a clean and simple way of representing conic sections. The mechanism used for defining conic sections is based on earlier work carried out by both Forrest and Pavlidis in [4]. Pavlidis, apart from highlighting the fact that conics have a longer and solid historical background, has noted that one of the main benefits of using conics is it is much easier to find the intersection of a line with a conic than with a cubic. The need of this is felt in many applications of Computer Graphics, like hidden surface removal.

In short, it can be seen that most of the advantages offered by Bezier cubic splines can equally be matched by quadratic splines. This research work will be oriented towards modeling of objects by using conic splines and trying to improve the smoothness of such curves.

1.1 Theory of Splines

The techniques for obtaining a mathematical curve model from digitized data is generally referred to as *curve fitting* technique. The technique by which the mathematical description of space curve is generated without any prior knowledge of the curve shape or form is referred to as *curve fairing* technique. This curve fairing technique is characterized by the fact that the few, if any, points on the curve pass through the control points used to define the curve.

- **Definition of spline** –A mathematical spline is a piecewise polynomial of degree K with continuity of derivatives of order K-1 at the common joints between segments.

Piecewise splines of low-degree polynomials are most useful for curve fitting because low degree polynomials reduce numerical instabilities that cause undesirable oscillations when several points are joined in a common curve.

Since low-degree polynomials cannot span an arbitrary series of points, a solution to overcome this is to use adjacent polynomial segments. A common technique is to use a series of cubic spline segments with each segment spanning only two points [5]. Also, the cubic spline is advantageous since it is the lowest degree curve which allows a point of inflection and has the ability to twist through space. The shape of the segment depends on the position and the tangent vectors at the ends of the segment. However, to represent a complete curve, multiple segments have to be joined together.

For a given *bounded curve* $F(u)$, $u\epsilon[u_0, u_1]$, $\phi_F = [\theta_1, \theta_2]$ is defined as the included tangent angle range, where θ_1 is the minimum (or maximum) of the slope angle of all the tangents to the bounded curve $F(u)$, θ_2 is the maximum (or minimum), and for all $\theta\epsilon[\theta_1, \theta_2]$, the curve has at least one point with that tangent angle.

A bounded curve $F(u)$, $u\epsilon[u_0, u_1]$ is defined to be a *C-shaped curve* if it satisfies the following conditions :

- $\phi_F < \pi$,
- the tangent slope angle varies in a slightly monotonic manner as u varies from u_0 to u_1.

For a *C-shaped* curve the following properties must be noted :

- the minimum and maximum of the tangent angles occur at the end points.
- for any angle value i.e., the slope given by say (dx/dy), there exists atmost one point whose tangent angle is equal to this value.
- The convex quadrilateral bounding the C-shaped curve is given by the quadrilateral formed by the two end tangent directions and a tangent to the curve parallel to the chord. Further, two disjoint triangles enclosing the curve are also defined.

By solving the quadratic equation $dx(F_y'(u)) = dy(F_x'(u))$, the parameter value of the point with a given tangent angle can be determined.

Also, to find the common tangents, the parametric range of both curves is made the same i.e., for $F(u)$ and $G(u)$, where $u\epsilon[u_0, u_1]$. This is called **reparameterization** of curves.

If $F(u)$ and $G(u)$, $u\epsilon[u_0, u_1]$, are two curve segments such that $\phi_F = \phi_G$, then let $F'(u_0) = \alpha G'(u_0)$ where α is some constant. The directions of the two curve segments is found from the value of α.

- If $\alpha > 0$, the curve segments are in *same direction* and
- If $\alpha < 0$, the two curve segments are in *opposite directions.*

1.1.1 Curvature Continuity Properties

The continuity properties exhibited by the curve segments may change depending on the way the segments are joined. G^0 geometric continuity is obtained by simply joining two segments together at a common end point. If the tangent vectors to each curve segment at the common end point match to within a constant by having their directions equal but unequal magnitudes G^1 geometric continuity is exhibited by the curve. If the tangent vectors match both in the direction and magnitude, the C^1 or first degree continuity is exhibited. In general for a curve $\mathbf{Q}[\mathbf{u}]$, if the direction and magnitude of $\frac{d^n}{dt^n}$ $[\mathbf{Q}(\mathbf{u})]$ are equal at the join point between segments then the curve exhibits C^n continuity.

Consider a parametrically defined curve given in two dimensions by $Q(t) = (Q_1(t), Q_2(t)), a \leq t \leq b$. If the curve is continuous and if the unit tangent vector to the curve is also continuous we say the curve has first-degree geometric continuity G^1. If in addition to the previous condition the curvature vector of the curve is also continuous then we can say the curve has second-degree geometric continuity G^2. For the curve to be visually smooth, it is essential only that it be G^1 continuous, and desirable that it be G^2 continuous.

Then it becomes important that the tangent vector at the internal joint between the two segments is known [6]. Since a cubic spline has a *second-order continuity* at the internal joints, the curvature (the second derivative of the curve) at the end of the first segment and the curvature at the beginning of the second segment is equal. By equating these two curvatures the tangent vector at the internal joint between any two spline segments can be found. Then for each piecewise cubic spline segment the end position and the tangent vectors for that segment are used to determine the blending functions for that segment. This *blending functions* are used to scale the control points into a curve segment. Any point on the cubic spline segment is a weighted sum of the end positions and the tangent vectors [7].

1.1.2 Minimizing Curvature / Maximizing Smoothness

The significant difference in magnitude of the blending functions shows that in general the end position vectors have relatively more influence than the end tangent vectors. Each curve segment shares control points whereby C^2 continuity is maintained between curve segments.

- At the position where there is no knot all basis functions are active and sum to unity.
- When a knot value is reached one basic function becomes inactive and another active.

- A B-spline curve with $m + 1$ control vertices is made up of $m - 2$ segments defined by the position of $m + 1$ basis functions in m, then the basis functions sum to unity in the range $m - 3$ to m, the values of the parameter u over which the curve is defined. At the knot value there are $m - 3$ basis functions that sum to unity.

- In case of uniform cubic B-spline each basis function is a cubic composed itself of four segments. Since a piecewise cubic spline curve is determined by the position vectors, tangent vectors and the end values of the parameter at the end of each segment.

To minimize curvature and hence maximize smoothness, the blending functions that effect the middle part of the segments must be minimized for each segment by choosing appropriate values of the parameter at the end of each segment.

In general, a B-spline curve does not interpolate any control points. However by introducing multiple vertices, a B-spline curve can interpolate control points but this involves a loss of continuity. The multiple vertices can be introduced anywhere, at the end or at an intermediate position.

When a control point is *repeated*, its influence will increase on the curve causing the curve to get pulled towards the repeated point. Since a segment is made by basis functions scaling control points and if a control point is repeated it will be used more than once in the evaluation of a single segment. As the *multiplicity* of the control points increases the continuity changes from C^2G^2 to C^2G^1, from C^2G^1 to C^2G^0. As the continuity changes the smooth curves become straight lines on either side of the repeated point.

However, the control points can be interpolated by another method in which the parametric intervals between successive knot values need not be equal, and most often these intervals are reduced to zero by inserting *multiple knots.* This method does not have the undesirable effect of straight-line curve segments on either side of the control point which is interpolated by the curve. It is very important to note that the knot sequence be non-decreasing so that successive knot values can be equal and between each successive knots there are four B-spline basis functions which consists of four segments over the five knot values.

If *uniform knot values* are used the basis functions are same for each parametric interval while with *non-uniform knot values* the basis functions are no longer the same for each interval, but rather vary from curve segment to curve segment.

To explain the effect of a multiple knot on the shape of a basis function consider that a uniform B-spline basis function is made up of four cubic polynomial segments defined over five knot values and are simply translations of each other in the parameter.

- If a knot at a particular position is *doubled* then that segment of the basis function which was defined between the doubled knot positions reduces to zero length and the function becomes asymmetric and it eliminates second derivative continuity C^2 with only first derivative continuity C^1 remaining.

- When a knot is *tripled* then two segments defined between the three knot positions (which are combined now into one by tripling) reduces to zero length leaving the function with positional continuity C^0, and similarly *quadrupling* a knot causes the function to lose even its positional continuity.

Thus *increasing the multiplicity of knots leads to the reduction in the number of segments.* It is not necessary that the spacing between knots is reduced to zero to have multiplicity, it can also be done that there is *non-uniform spacing* between successive knots with two knots set very close to each other and some very far. Thus, by varying the spacing between knots and adjusting the multiplicity between knots a designer can have greater control over the shape of the curve. Also an additional knot and control point can be added to non-uniform B-splines so that the resulting curve is easily reshaped. Though more complex curves can be represented by increasing the order of the polynomial, it has computational and mathematical disadvantages that are overcome by splitting the curve into segments and applying some constraints at the joints.

B-spline curves possess a number of properties that make them appropriate for CAD applications :

1) they approximate precisely straight line segments,

2) they preserve monotonicity and convexity,

3) they have *convex hull property* and *variation diminishing property*,

4) a change in one control point d_i, affects only the nearby segments of the curve i.e., those corresponding to the knot intervals (t_l, t_{l+1}), with $l = i-1, i, ..., i+2$.

5) the knot vector controls the degree of continuity between the polynomial segments : if t_i has multiplicity $r(i.e., t_i = t_{i+1} = ... = t_{i+r-1})$, then the corresponding B-spline of degree n is at least C^{3-r} at that knot.

6) the range of the parameter u is divided into $n + d$ subintervals by values specified in the knot vector.

7) the curve is transformed by applying any affine transformation to its control point representation.

1.2 Previous Work on Cubics

A piecewise approximation to a shape will be most efficient when as few pieces as possible are used and there is a trade-off between a tight fit and an efficient representation. The positions and smoothness of the joints between pieces is critical in getting a good representation of the shape but *finding the best knot position and tangents* at each non-corner knot is not an easy task.

In [8] a method of dynamic programming is used to find the knots. In the dynamic programming phase a single-piece fitting routine is used to measure how well a span of sample points may be fit with a single cubic piece, and after the knots are found, it is used to find the piecewise approximation. One simple method to define the knots is to 'grow' the cubic pieces out until the fit for that cubic piece exceeds some threshold. In other words, keep adding data points starting from previous knot until the piece is as long as possible with each piece constrained to maintain continuity.

But this method is vulnerable to local phenomenon so a new dynamic programming is applied for the best solution set of knots in a more global manner. If a point is identified as a corner a knot is put at that location and tangent continuity is not applied there.

Finding common tangents to planar parametric curves is another important issue which is described in [9]. In order to achieve greater control on the shape of the curve, which are defined in Hermite, Bezier or B-spline form, the curves are usually divided into a number of segments which exhibits the various types of continuities like C^2,GC^2,C^1....

If $F(u) = (F_x(u), F_y(u)), u \epsilon [u_0, u_1]$ and $G(v) = (G_x(v), G_y(v)), v \epsilon [v_0, v_1]$ are the polynomials defining the two curves then the points at which there are common tangents to the two curves are given by the values of the u, v which simultaneously satisfy the equations (1) and (2)

$$\frac{F_y(u) - G_y(v)}{F_x(u) - G_x(v)} = \frac{F_y'(u)}{F_x'(u)} \tag{1.1}$$

$$\frac{F_y(u) - G_y(v)}{F_x(u) - G_x(v)} = \frac{G'_x(v)}{G'_y(v)} \tag{1.2}$$

where F' and G' are obtained by differentiating once.

- The above two equations state that for some u and v, the line segment joining $F(u)$ and $G(v)$ is along the tangents at $F(u)$ and $G(v)$ respectively.

- F and G are of degrees n and F', G' are of degree $n - 1$; which makes each equation of order $n + (n - 1)$ in two variables. The pair of equations could possibly give rise to $(2n-1)^2$ different solutions and for two cubic curves there are 25.

- The number of tangent lines will depend on the actual shape characteristics of the two curves and their relative configuration.

One of the important tasks of a modern CAD system is to generate a *fair* or *visually pleasing* curve from given data points and it is quite possible that the resulting curve is not fair enough for some data sets and some processing of the geometry of the curve is required in order to obtain an acceptable curve. A curve is characterized as fair if the corresponding curvature plot is continuous, has the appropriate sign and is as close as possible to a piecewise monotone function with as few as possible monotone pieces.

The fairness of a B-spline curve is affected by the knot vector (parametrization) used. The problem consists of sharp corners (local extrema) that appear, almost always, at points of the curve corresponding to knot locations. A feature of a B-spline curve which is very important for this fairness issue is knot insertion where inserting a new knot $\tau(t_l < \tau < t_{l+1})$ into the knot vector results in increasing the control points by one. If the new spline S^τ(t) is to be identical to the original spline $S(t)$, where the new knot vector is $V^\tau = V \bigcup \tau$, the two control points must be replaced by three new ones determined by the following :

$$d_i^\tau = \frac{t_{i+2} - \tau}{t_{i+2} - t_{i-1}} d_{i-1} + \frac{\tau - t_{i-1}}{t_{i+2} - t_{i-1}} d_i, i = l-1, l, l+1 \tag{1.3}$$

The order of knot insertion is independent and for a finite number of new knots the resulting control polygon is independent of the other in which these knots are inserted.

Based on this knot-insertion a new method called *knot removal-knot insertion* is described in [10] for improving the fairness of a B-spline curve. Here, the smoothness of the curve is increased by increasing the continuity of the curve to the highest possible continuity at the knot where there is discontinuity.

A B-spline curve should always be faired at the offending knot location.

For a cubic B-spline curve s(t) with control polygon $C = \{d_0, ..., d_{L+2}\}$ and knot vector $V = \{t_0 \leq t_1 \leq ... \leq t_{L+4}\}$, the offending knot $t_j, (2 < j < L+2)$ is to be identified at which the spline has to be faired, and this offending knot can be multiple knots also. The fairing process then includes the two steps

1) knot removal: remove t_j from the knot vector and compute the new set of control

points $C = \{D_0, ..., D_{L+1}\}$ that correspond to a spline curve $s(t)$ with knot vector $V = V - \{t_j\}$.

2) knot reinsertion: reinsert t_j, so that, the 'faired' curve is defined over a knot vector but the corresponding control polygon $\bar{C} = \{\bar{d}_0, ..., \bar{d}_{L+2}\}$ is not identical to C.

The knot removal is based on the fundamental requirement that each control point of the faired curve $\bar{s}(t)$ must be as close as possible to the corresponding one of the initial curve $s(t)$.

This fairing scheme affects the curve by increasing the smoothness of the spline at t_j, from C^2 to C^3, that is, the faired curve has the highest possible continuity at t_j. For cubics C^3 continuity is equivalent to C^∞ continuity. This can be understood from the fact that the curve has C^{3-r} continuity at the knot t_j with multiplicity r. So, if the knot is removed from its offending position, then at that position its multiplicity is 0 which makes the continuity $C^{3-0} = C^3$.

In [11] an algorithm is presented which modifies the control points of a B-spline curve and makes the fixed curve strictly convex. Here the geometric shape in the vicinity of a point G_0 on a surface G is considered in terms of total curvature and depending on this curvature a set of points are modified iteratively until the curve is completely convex, that is, at each point the curve is checked for locally convex condition, with respect to a set of neighboring points. Since the iteration starts at one end point and ends at other end point, it is possible for the ending point of the new convex curve to deviate from that of the original curve with some error. By using a back-shift method all the points are moved so that the curve maintains its convexity and at the same time does not deviate from its original end points.

Additional degrees of freedom can be introduced to a curve either by *subdivision* or by *degree raising.* In the subdivision approach the idea is to split the parameter interval and represent the curve by two separate Bernstein polynomials with a common value at the mid-point as discussed in [9] and also new knots can be added at the appropriate points where smoothness is required as done in [10].

The degree-raising approach is based on the fact that a polynomial of order m can be written in terms of Bernstein basis functions of order $m+1$ and raising the degree by one adds one extra degree of freedom to the representation. If the original spline had knots located at k different interval points, then raising the degree by one increases the number of degrees of freedom by $k-1$. In [12] stable efficient algorithms are presented to raise the degree of curves (or surfaces) so that a curve written as a linear combination of m^{th} order B-splines is rewritten as a linear combination of $m+1^{st}$ order B-splines with a minimal number of knot insertions. It is also shown that as the degree of splines increases, the control polygons associated with a given curve converge to a curve.

While the parametric form of curves is studied extensively, the paper [13] studies splines created from segments of algebraic curves which are given in implicit form. Each segment is chosen to interpolate one more point and slope and has two additional fullness parameters to control the shape. The method described in [13] defines a control polygon as a sequence of triangles, an interpolation point is selected inside each triangle and a tangent line at each interpolation point is selected. Then by imposing different conditions on the coefficients of the implicit equation and selecting different interpolation points conic as well as cubic splines are created. In [14] by performing shape-matching task in which a given curve is manipulated to match a tangent curve precisely, the performance measures among some formulations of cubic splines is investigated.

By selecting seven design points it was found that B-spline curve contained four segments joined together with C^2 continuity, a bezier curve contained two segments joined together with C^0 continuity with each segment intersecting the first and the last of the four successive design points defining the segments, a Catmull-Rom curve contain four segments joined together with C^1 continuity with all but the first and last design points lying on the curve, an end-condition curve contained four segments joined together with C^2 continuity and a natural curve has six segments with C^2 continuity and curve interpolates all design points.

For Catmull-Rom and End-Condition curve the first and last of design points which does not lie on the curve influence the curve's position, tangent and curvature at initial and final points of the curve. The end-point and natural spline formulation both exhibit global control while the remaining spline formulation can exhibit local control. With cubic B-splines as a generalization Barsky [15] introduced cubic Beta-splines which provided the means of constructing parametric spline curves in which the shape of the curve is controlled by bias parameter β_1 and tension parameter β_2. In the paper [16] Goodman gave an explicit formula for cubic beta-splines on a uniform knot sequence with varying β_1 and β_2 values at the knots.

A cubic B-spline curve has second-degree parametric continuity, whereas a cubic Beta-spline curve has second-degree geometric continuity, that is, continuity of the curve, the unit tangent vector, and the curvature vector. In [15] Barsky used a uniform knot sequence and constant β_1 and β_2 values at all knots or joints of the curve and introduced the extension where β_1 and β_2 can be varied to provide local control of bias and tension in the curve.

This idea was later presented in [17] in which Hermite interpolation is used to give continuous β_1 and β_2 values throughout the curve. Then in paper [18] the Beta-splines have been further extended by the introduction of nonuniform knot sequences and varying β_1 and β_2 values at the knots without using interpolation. In [19] an explicit formula for cubic Beta-Splines on a nonuniform knot sequence with constant $\beta_1 = 1$ and varying β_2 values at the knots is developed which can be used if the knot sequence contains multiple knots which is useful for knot insertion. By applying linear transformation of a nonuniform distinct knot sequence with varying β_1 and β_2 values at the knots a uniform knot sequence with different β_1 and β_2 values can be obtained. In [20] the effects on the Beta-splines curve is studied when β_1 and β_2 values at the knots are varied. In this paper, by assuming the control vertices and any parameter to be fixed β_1 and β_2 are varied only at internal knots (that is $\beta_1(j)$ and $\beta_2(j), 2 \leq j \leq k-3$ where $k+1$ is the number of control vertices), some of the following observations were made.

- Case 1 : $\beta_2(j) \longrightarrow \infty$

  When β_2(j) is let to be large, a point on the design curve is pulled toward the control vertex V_j and if both β_2(j) and β_2(j+1) were let to be large, a section of the design curve is pulled toward straight line from V_j to V_{j+1}.

- Case 2 : $\beta_1(j) \longrightarrow \infty$

  Letting β_1(j) become large pulls a section of the design curve toward a section of the straight line from V_{j-2} to V_{j-1}, including V_{j-1}. Compared to β_2(j) becoming large this effect is less 'local'.

- Case 3 : $\beta_2(j) \longrightarrow \infty$ and $\beta_1(j) \longrightarrow \infty$

  Letting both $\beta_1(j)$ and β_2(j) become large pulls a point on the design curve toward a point on the straight line from V_{j-1} to V_j, the relative distances of this point from V_{j-1} and V_j depends on the relative magnitudes of β_2(j) and β_1(j).

Letting β_1(j) tending to zero pulls a point on the design curve toward a point on the straight line from V_j to V_{j+1} and the point moves from V_{j+1} to V_j as increases. When β_1(j) becomes close to zero and β_2(j) is zero the curve's section is pulled toward a section of the straight line from V_{j+1} to V_{j+2}.

The parametric and geometric continuity of any rational curve is ensured by requiring the homogenous polynomial curve associated with the rational curve to possess either parametric or geometric continuity respectively [21]. A rational function is a scalar function $r : R \longrightarrow R$ which can be expressed as $r(u) = \frac{f(u)}{g(u)}$, where f and g are polynomials in u. A polynomial curve is a vector-valued function, each component of which is a polynomial and a rational curve is a vector-valued function, each component of which is a rational function.

A rational curve q can be thought of as a vector-valued function, each component of which is a rational function, or q can be thought of as the composition of a vector-valued polynomial function $Q : R \longrightarrow R^{d+1}$ with a projection function that takes $(x_1, \cdots, x_d, x_{d+1})$ to $(\frac{x_1}{x_{d+1}}, \cdots, \frac{x_d}{x_{d+1}})$.

Piecewise cubic parametric curves with a slightly different continuity class of GC^2 is discussed in the paper [22]. The ν-spline curve is used as an alternative to the spline in tension since these curves are piecewise hyperbolic sines and cosines which makes computation expensive. In ν-splines when two adjacent tension parameters are increased to apply tension to a curve segment, the individual component functions do not converge to linear functions as they do for splines in tension.

In [23] a spline for the representation of parametric curves in both interpolatory and B-spline form is used and the rational spline is not restricted to the 'homogenous coordinates' form of having a C^2 cubic spline numerator and denominator and in general it is not a projection from a C^2 cubic spline in R^4 into R^3 which is the case of non-uniform rational B-splines. This rational spline provides an alternative to the cubic ν-spline of [22]. In order to be computationally more economical, the curve method presented in [24] requires only quadratic denominator and with two shape parameters in one interval different kinds of shape effects like biased point and interval tensions can be achieved in a well-controlled manner. In [25] weighted ν-spline which is a C^1 piecewise cubic polynomial interpolant is discussed. This weighted ν-spline generalizes cubic splines, weighted splines and ν-splines. The weighted ν-spline has shape controls to tighten the curve on segments between interpolation points and/or tighten the corners of the curve at interpolation points. Depending on the values of the shape control parameters the curve is a C^2 spline or a weighted spline or a ν-spline.

1.3 Proposed Work

1.3.1 Motivation

In this work, low-degree piecewise rational parametric polynomials are used to represent curves and objects. The reasons for selecting this type of representation can be briefed as follows :

- A piecewise curve of low-degree polynomial is taken because they are most useful for curve fitting as low-degree polynomials both **reduce the computational requirements and also reduce the numerical instabilities** that arises with higher degree curves. These instabilities cause undesirable oscillations when several points are joined in a common curve.

- Since low-degree polynomials cannot span arbitrary series of points, **adjacent polynomial segments are used**. A common technique is to use a series of segments with each segment spanning only two points.

- A parametric representation allows a curve to be represented as a set of piecewise polynomials so that **complex curves can be designed**. Applying a linear transform to the parametric representation of the curve, transforms the curve itself.

- A rational curve is invariant under rotation, scaling, translation and perspective transformations of the control points whereas non-rational curves are invariant under only rotation, scaling and translation. By applying the perspective transformation to the control points the perspective transformation of the original curve is obtained. **Also rational curve can define precisely any of the conic sections.**

1.3.2 Problem statement

The objective of this research is to develop the rational quadratic form of a curve which can approximately represent any rational cubic curve with negligible difference. It should be noted that in this work it is shown how a conic can be used as an alternative to a rational cubic and it is different from converting a rational cubic into a conic as done in [26]. Though a conic is shown along with rational cubic, it is done just to show the similarity of both representations and it does not mean that one should first draw a rational cubic in order to get a conic. Both can be drawn separately.

The main objectives of this research are :

- To implement a method for obtaining an interpolatory C^1 (tangent) continuous conic which can be used as an alternative to an interpolatory C^1 (tangent) continuous rational cubic.

- To implement a method for obtaining an interpolatory C^2 (curvature) continuous conic which can be used as an alternative to an interpolatory C^2 (curvature) continuous rational cubic.

- A comparison analysis of the interpolatory rational cubic form and the interpolatory conic is made.

- The approximation form of the conic obtained from rational cubic is also investigated. This is an improvement of the work done in [27].

- A comparison analysis of the approximatory rational cubic form and the approximatory conic is made.

- Then, the smoothness of the interpolatory and approximatory conic developed in previous steps, is improved.

- A comparison analysis based on the smoothness effect is made between the rational cubic form and the smoothed interpolatory and approximatory conics obtained from previous step.

- Also, various ways of drawing a conic and different methods of obtaining offsets for conics is implemented.

The foremost objective of this research is to show that, to design a tangent or curvature continuous curves it is not always necessary to use piecewise cubics. Though rational cubic is the lowest-degree polynomial which allows a point of inflection and has ability to twist through space, conic segments can also be pieced together to form curves that are tangent or curvature continuous. Here, basically a rational cubic is split at some point (usually at the mid-point) and represented by two conics so that the problem of representing inflection points in conics can be overcome. It is also shown that all the properties of rational cubic can well be satisfied by conics. The above curve methods are implemented for both 2-D and 3-D object modeling.

Chapter 2

Different ways of defining rational quadratics

2.1 Literature Review

Traditionally conic segments were pieced together to form curves with tangent continuity. In their papers [3] and [4] Pratt and Pavlidis described how to form a conic segment from curves with tangent continuity. Later in the paper [28] Farin presented a construction of a conic spline with curvature continuity, by utilizing the geometry underlying the Bezier form. In [1] Farin developed a subdivision algorithm for curves and in [28] by using this subdivision algorithm in conjunction with reparametrization, developed a simple recursive method to find the offset of a conic section.

Offset curves are frequently used in designing fonts but since the offset of a conic is not usually a conic an approximation method is developed. If C is a given curve, $\bar{C}(t) = C(t) + dn(t)$ defines an offset curve where d is the offset distance and $n(t)$ is the normal of C, pointing in the desired offset direction.

A tangent continuous piecewise rational quadratic i.e., a conic spline does not need the notion of a knot sequence for its definition. For every open conic spline a knot sequence can be found over which it becomes a differentiable piecewise rational quadratic. The control polygon must satisfy the consistency condition in order to find a knot sequence over which the curve is a differentiable piecewise quadratic.

A general conic arc is derived by a quadratic transformation using an appropriate parabola as a basic curve in [29]. The transformation is defined purely by means of geometric data and a double ratio.

In [30] the construction of curvature continuous, locally convex conic splines is discussed where the elements of the spline consist of two conic arcs pieced together with second- or third- order geometric continuity. In this paper a Hermite-like curvature continuous elements, consisting of two conic arcs which allow to interpolate two points where tangents and curvatures at the points are given.

These G^2 conic elements possess a simple mathematical description and several useful interesting geometric properties which are appropriate for the design of open or closed curvature continuous planar curves but since the generated curves do not possess points with vanishing curvature they are always locally convex and never have inflection points.

A new condition is derived for NURBS to represent conics precisely in the paper [31]. Also the necessary and sufficient conditions for the representation of a circular arc using the full circumscribing square-based representation using only four vertices which is less than the number required with the other methods which requires at least seven control vertices. Here it is showed that to represent a circular arc using NURBS of degree two there are no constraints on the control polygon and the end points of the arc need not coincide with the control vertices.

The type and shape of a Conic section is determined by a parameter termed *sharpness*. As any conic section can be achieved by combining a single control point where the two endpoint tangents intersect and a 'curve weighted' parameter (sharpness value), it is an advantage to use conics rather than cubic splines.

To prefer conics the two main properties of curvature continuity and zero curvature at inflection points offered by cubics are to approximated as in [26]

- **Curvature continuity using Conics** By using the fact that as the sharpness value goes from zero to infinity the corresponding curvature at the endpoints goes from infinity to zero, a formula is derived for evaluating curvature at the endpoints of a conic curve.

- **Zero curvature at inflection points** As points of inflection are non-existant in conics, a method to represent such points using conics was developed by using a formula to make the curvature at the inflection point much smaller than any prevailing curvature at some distance away from the inflection point which enables piecewise conic segments to approximate zero curvature at inflection points.

To decide a method to recover the outlines described by Bezier cubic curve segments two possibilities are to be considered

- In the case when two endpoint tangents do not intersect, the Bezier curve needs to be subdivided into two halves, with two different conics representing each half. A measure of quality of the conic-fit needs to be made at different splitting points along the Bezier curve to effeciently recover Bezier cubic curves.

- In the case when the two endpoint tangent do intersect the need to subdivide the Bezier curve depends on whether a single conic approximation is within a desired best-fit criterion.

The problem of fitting the best conic(s) to a given Bezier cubic curve can be considered as a general problem of fitting the best curve to a given set of data points. The approach used in [26] uses a least-squares method which is of linear type and it encompasses the basic definition of conics presented by Pratt (1985).

The basis of the technique starts with the fundamental quadratic equation that describes all conic sections, which takes the form d=0, where

$$d(x,y) = 2vx - 2uy - \alpha y^2 - \beta x^2 - 2\gamma xy + c$$

The control point is used to define the initial gradient $\frac{v}{u}$, so, by taking advantage of the arbitrary scale, the control positions u and v are set equal to the x and y coordinates of the control points respectively. The gradient at any point on the conic curve can be expressed as:

$$\frac{dy}{dx} = \frac{v - \beta x - \gamma y}{u + \gamma x + \alpha y}$$

Since the control point also defines the gradient at the endpoint, the corresponding gradient at the end point whose x and y coordinates be, say, C_x and C_y respectively, will take the form

$$\frac{v - \beta C_x - \gamma C_y}{u + \gamma C_x + \alpha C_y} = \frac{C_y - v}{C_x - u}$$

By using the conic equation $d(C_x, C_y) = 0$ and the respective gradient equation at the endpoint, an expression for u and v can be realised in terms of α, β, γ, C_x and C_y as follows

$$u = \frac{C_x - \gamma C_x - \alpha C_y}{2}$$

$$v = \frac{C_y + \beta C_x + \gamma C_y}{2}$$

Substituting these findings, the general conic form becomes

$$d(x, y) = \alpha(yC_y - y^2) + \beta(xC_x - x^2) + \gamma(xC_y + yC_x - 2xy) + (xC_y - yC_x)$$

For each Bezier cubic curve segment, described in terms of its endpoints and two control points, the following parametric form is used to obtain the x and y positions

$$x(u) = x_0(1 - u)^3 + 3x_1u(1 - u)^2 + 3x_2u^2(1 - u) + x_3u^3$$

$$y(u) = y_0(1-u)^3 + 3y_1u(1-u)^2 + 3y_2u^2(1-u) + y_3u^3$$

where u ranges from 0 to 1, x_0, y_0 and x_1, y_1 denote the start and endpoints repectively, x_2, y_2 and x_3, y_3 are the respective control points.

To generate, therefore, a Bezier curve consisting of some data points the parameter u has to be incremented at each point and each data point, in turn, is then substituted in the equation of conic to gain a corresponding residue value $d(x_i, y_i)$. By doing this simultaneous equations are set-up one for each data point to solve for three unknowns α, β and γ. The solution of these unknowns is such that it minimises the summation of the squared residue at each data point, that is

$$\frac{\partial}{\partial\alpha}\sum_{i=0}^{p} d^2(x_i, y_i) = 0$$

$$\frac{\partial}{\partial\beta}\sum_{i=0}^{p} d^2(x_i, y_i) = 0$$

$$\frac{\partial}{\partial\gamma}\sum_{i=0}^{p} d^2(x_i, y_i) = 0$$

where p is the number of data points, so that the "best" possible conic section is used to approximate the given Bezier spline.

Once the values for α, β and γ are found, the corresponding control positions u and v are evaluated by using their respective equations. The sharpness value **S** for this conic section is known by using the expression

$$\alpha\beta - \gamma^2 = \frac{1}{S^2} - 1$$

so that

$$S = \frac{1}{\sqrt{1 + \alpha\beta - \gamma^2}}$$

At each Bezier splitting point, since two conic sections are fitted, the total residue from both conics is used to assess the best point to split which ensures that conics fitted to both sides of the splitting point are those that return the overall minimum residue value.

In [27] a C^1 continuous approximatory rational cubic and conic were developed. It was shown how a C^1 continuous rational cubic can be alternatively represented by a conic.

Many equivalent ways exist to define a conic section as explained in [32] and [28]. For our purpose the following one is very useful:

2.2 Standard form

A conic section in $\mathbf{E}^2$ *is the projection of a parabola in* $\mathbf{E}^3$ *into a plane* as shown in [32]. It is useful to abandon the principle of being independent of a fixed coordinate system when the conics are formulated as rational curves and the center of the projection is chosen to be the origin 0 of a $3D$ Cartesian coordinate system. The plane $z = 1$ is taken to be the plane into which one projects and it can be thought of as a copy of $\mathbf{E}^2$, thus identifying points $[x\,y]^T$ with $[x\,y\,1]^T$. Our special projection is characterized by

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} x/z \\ y/z \\ 1 \end{bmatrix}$$

Note that a point $[\mathrm{x} \quad \mathrm{y}]^T$ is the projection of a whole family of points: every point on the straight line $[\mathrm{wx} \quad \mathrm{wy} \quad \mathrm{w}]^T$ projects to $[\mathrm{x} \quad \mathrm{y}]^T$. In the following discussion, the shorthand notation of $[\mathbf{wx} \quad \mathrm{w}]^T$ with $\mathbf{x} \in \mathbf{E}^2$ is used for $[\mathrm{wx} \quad \mathrm{wy} \quad \mathrm{w}]^T$.

Let $\mathbf{c}(t) \in \mathbf{E}^2$ be a point on a conic. Then there exist numbers $w_0, w_1, w_2 \in \mathbf{R}$ and points $b_0, b_1, b_2 \in \mathbf{E}^2$ such that

$$\mathbf{c}(t) = \frac{w_0\mathbf{b}_0B_0^2(t) + w_1\mathbf{b}_1B_1^2(t) + w_2\mathbf{b}_2B_2^2(t)}{w_0B_0^2(t) + w_1B_1^2(t) + w_2B_2^2(t)}; 0 \leq t \leq 1 \tag{2.1}$$

Now, the above equation will be proved. The point $c(t) \in \mathbf{E}^2$ can be identified with $[\mathrm{c(t)} \quad 1]^T \in \mathbf{E}^3$. This point is the projection of a point $[\mathrm{w(t)c(t)} \quad \mathrm{w(t)}]^T$ which lies on a 3D parabola. The third component $w(t)$ of this 3D point must be a quadratic function in t, and may be expressed in Bernstein form

$$w(t) = w_0B_0^2(t) + w_1B_1^2(t) + w_2B_2^2(t)$$

Now, after determining $w(t)$, it can be written as

$$w(t)\begin{bmatrix} c(t) \\ 1 \end{bmatrix} = \begin{bmatrix} c(t)\Sigma w_iB_i^2(t) \\ \Sigma w_iB_i^2(t) \\ 1 \end{bmatrix}$$

Since the left hand side of this equation denotes a parabola, we may write

$$\sum_{i=0}^{2} \begin{bmatrix} \mathbf{p}_i \\ w_i \end{bmatrix} B_i^2(t) = \begin{bmatrix} \mathbf{c}(t)\Sigma w_i B_i^2(t) \\ \Sigma w_i B_i^2(t) \end{bmatrix}$$

with some points $\mathbf{p}_i \in \mathbf{E}^2$. Thus

$$\sum_{i=0}^{2} \mathbf{p}_i B_i^2(t) = c(t) \sum_{i=0}^{2} w_i B_i^2(t)$$

and hence

$$\mathbf{c}(t) = \frac{\mathbf{p}_0 B_0^2(t) + \mathbf{p}_1 B_1^2(t) + \mathbf{p}_2 B_2^2(t)}{w_0 B_0^2(t) + w_1 B_1^2(t) + w_2 B_2^2(t)}; 0 \leq t \leq 1$$

Setting $\mathbf{p}_i = w_i \mathbf{b}_i$ now proves the equation (2.1).

Here, the points $\mathbf{b}_i$ are called the *control polygon* of the conic; the numbers w_i are called *weights* of the corresponding control polygon vertices. The straight line segments $[b_0, b_1]$ and $[b_1, b_2]$ are tangents to $\mathbf{c}$ at $\mathbf{c}(0) = \mathbf{b}_0$ and $\mathbf{c}(1) = \mathbf{b}_2$, respectively. Thus the conic control polygon is the projection of the control polygon with vertices $[w_i \mathbf{b}_i \; w_i]^T$, which is the control polygon of the $3D$ parabola that was projected onto $\mathbf{c}$.

The form of`equation (2.1) is called the *rational quadratic form* of a conic section. If all weights are equal, we get nonrational quadratics that is parabolas. The influence of the weights on the shape of the conic is illustrated in *Figure* 2.1.

In the *Figure* 2.1 the $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_2}$ are the given control points which form a control polygon. The weights w_0, w_1, w_2 given for these curves are as follows:

curve 1 (consists of two coinciding curves): one curve has $w_0 = 2.0$, $w_2 = 12.0$, w_1

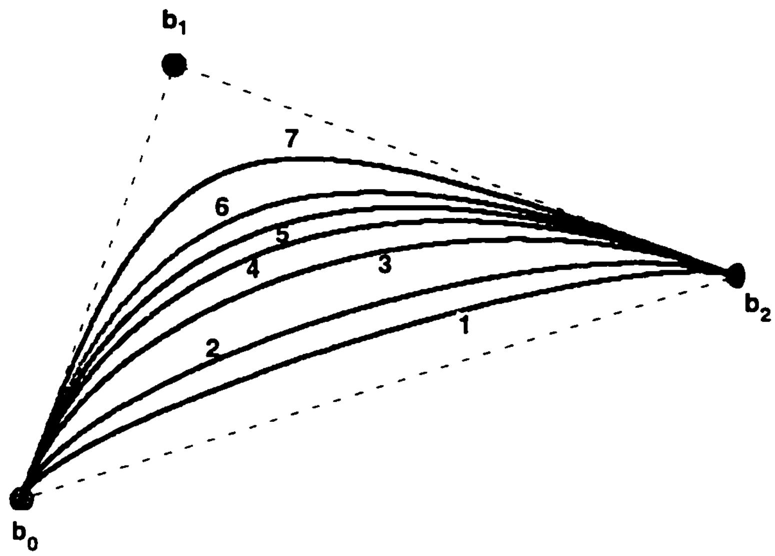


Figure 2.1: A rational quadratic curve

$= 0.8$, the other curve has $w_0 = 12.0$, $w_2 = 2.0$, $w_1 = 0.8$;

curve 2: $w_0 = 0.1$, $w_2 = 4.0$, $w_1 = 0.8$;

curve 3 (consists of two coinciding curves): one curve has $w_0 = 1.0$, $w_2 = 2.0$, $w_1 = 0.8$, the other curve has $w_0 = 2.0$, $w_2 = 1.0$, $w_1 = 0.8$;

curve 4: w_0, w_2 are not taken, w_1 is 0.8 ;

curve 5: all the three weights are equal($= 0.8$);

curve 6: $w_0 = 2.0$, $w_2 = 4.0$, $w_1 = 0.8$;

curve 7 (consists of two coinciding curves): one curve has $w_0 = 0.2$, $w_2 = 0.7$, $w_1 = 0.8$, the other curve has $w_0 = 0.4$, $w_2 = 1.4$, $w_1 = 1.6$.

As can be seen from the *Figure* 2.1, as the weights tends to increase and reach infinity the conic gets flattened and finally becomes a straight line and when the weights tends to decrease and approach zero value the conic becomes more and more attracted towards the triangle formed by the control points. Also note that when the three weights are same in any order for two conics, then those two conics will have the same shape as the curves number 1 and 3 shows it by coinciding with each other. A common nonzero factor in the w_i does not affect the conic at all as the *curve* 7 proves it by coinciding with another curve whose weights are multiplied by a common factor. If $w_0 \neq 0$, one may therefore always achieve $w_0 = 1$ by a simple scaling of all w_i. There are other changes of the weights that leave the curve shape unchanged: these correspond to *rational linear parameter transformations.* Suppose two rational quadratic curve segments possess the same control polygon but different weights. Denote the weights of one curve by w_i and those of the other one by $\hat{w}_i$. If the graphs of both curves are identical, one must be able to reparametrize one to obtain the other. Then the weights of both curves must be related by

$$\frac{w_0\hat{w}_1}{w_1\hat{w}_2} = \frac{w_1\hat{w}_0}{w_2\hat{w}_1} \tag{2.2}$$

If a rational quadratic is not in standard form, we may use this formula to rewrite it in standard form. This process is called *standardization.*

Let

$$t = \frac{\hat{t}}{\hat{\rho}(1-\hat{t})+\hat{t}}, (1-t) = \frac{\hat{\rho}(1-\hat{t})}{\hat{\rho}(1-\hat{t})+\hat{t}}$$

By substituting these into equation (2.1) we obtain

$$\hat{\mathbf{c}}(t) = \frac{\hat{\rho}^2 w_0 \mathbf{b}_0 B_0^2(\hat{t}) + \hat{\rho} w_1 \mathbf{b}_1 B_1^2(\hat{t}) + w_2 \mathbf{b}_2 B_2^2(\hat{t})}{\hat{\rho}^2 w_0 B_0^2(\hat{t}) + \hat{\rho} w_1 B_1^2(\hat{t}) + w_2 B_2^2(\hat{t})}$$

Thus, the curve shape is not changed if each weight w_i is replaced by $\hat{w}_i = \hat{\rho}^{2-i} w_i$. If, for a given set of weights w_i, we select $\hat{\rho} = \sqrt{\frac{w_2}{w_0}}$, then $\hat{w}_0 = \hat{\rho}^2 w_0$ which is equal to w_2 ; $\hat{w}_1 = \hat{\rho} w_1$; $\hat{w}_2 = w_2$. We obtain $\hat{w}_0 = w_2$, and after dividing all three weights through by w_2, we even have $\hat{w}_0 = \hat{w}_2 = 1$. A conic that satisfies this condition is said to be in *standard form.* All conics with $w_0, w_2 \neq 0$ may be rewritten in standard form with the above choice of $\hat{\rho}$.

The rational quadratic in standard form can be written as:

$$\mathbf{c}(t) = \frac{\mathbf{b}_0 B_0^2(t) + w\mathbf{b}_1 B_1^2(t) + \mathbf{b}_2 B_2^2(t)}{B_0^2(t) + B_1^2(t) + B_2^2(t)}; 0 \le t \le 1 \tag{2.3}$$

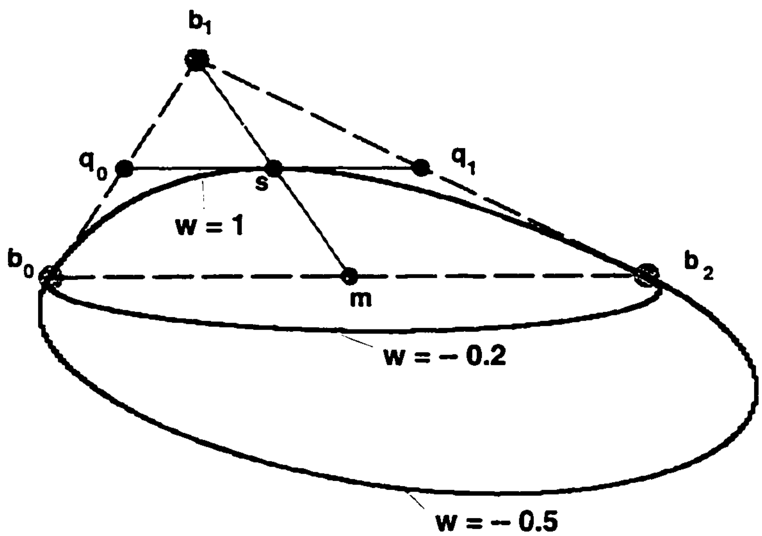


Figure 2.2: Effect of the weight parameter $w(< 0)$ on the quadratics is shown

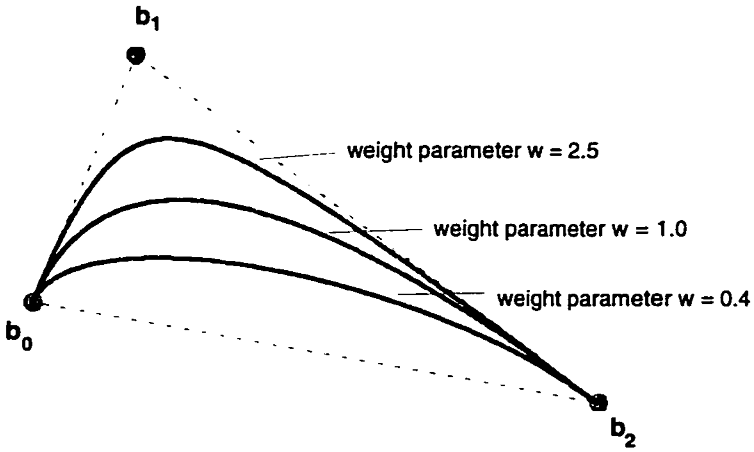


Figure 2.3: Effect of the weight parameter w on the quadratics is shown

The curve segment of equation (2.3) describes a segment of a conic section. We will therefore treat the terms *rational quadratic* and *conic* as being interchangeable. As can be seen from the *Figure* 2.3, for $w < 1$ an ellipse is obtained ; for $w = 1$ a parabola is obtained ; for $w > 1$ a hyperbola is obtained. As can be seen from the *Figure* 2.2, for $w \geq 0$, the conic of equation (2.3) lies in the *convex hull* of the control polygon ; for $w < 0$, the conic lies outside the *convex hull* of the control polygon.

It is possible to create a *circular arc* by assigning $w = sin\frac{\alpha}{2}$, where α is the angle at $\mathbf{b}_1$ if the control polygon forms an isosceles triangle with base $[\mathbf{b}_0, \mathbf{b}_2]$. More generally, for an arbitrary control polygon, the ellipse with minimum eccentricity is obtained by setting

$$w = \frac{l_{02}}{\sqrt{2(l_{01}^2 + l_{12}^2)}} \tag{2.4}$$

where $l_{ij} = ||\mathbf{b}_i - \mathbf{b}_j||$.

As can be seen from *Figure* 2.4 when the entered weight $w = 0.5$, the w for minimum eccentricity is also 0.5 and both the curves are identical. When the entered weight w is $0 \leq w \leq 0.5$ the curve is below the curve with minimum eccentricity, that is, it is more flatter. However, when the entered weight w is $w \geq 0.5$, the curve is above the curve with minimum eccentricity curve and is attracted towards the control point $\mathbf{b}_1$ and also the weight for minimum eccentricity starts decreasing from around 0.5 towards 0 as weight w increases towards infinity.

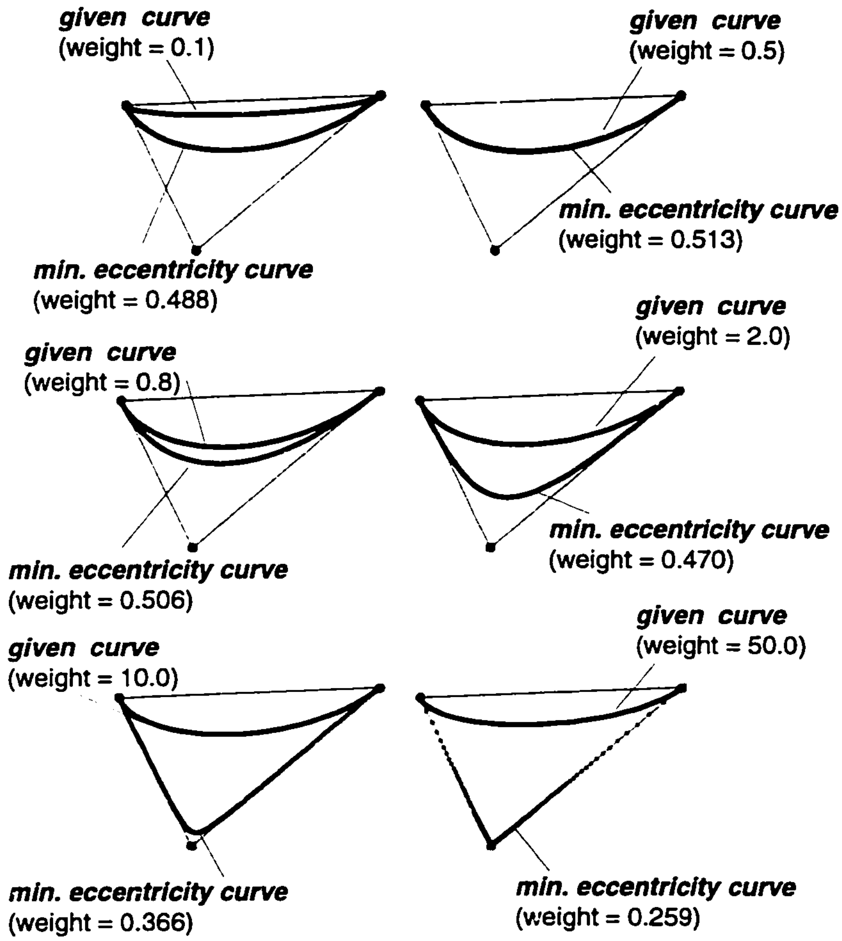


Figure 2.4: A rational quadratic curve and its minimum eccentricity curve is drawn

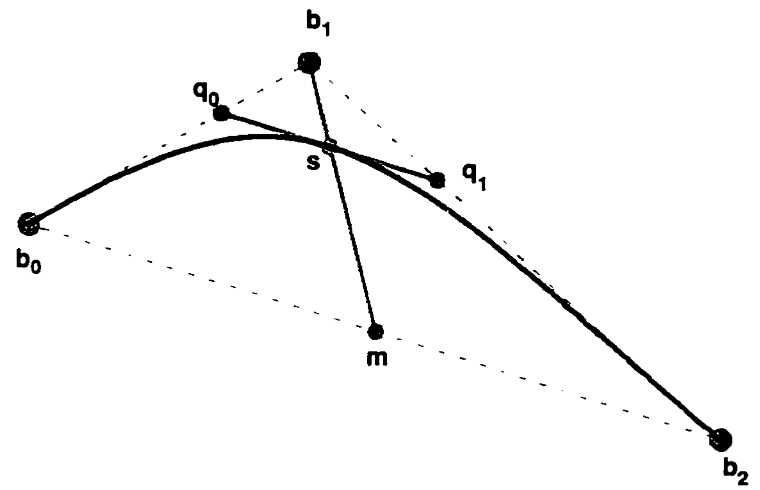


Figure 2.5: A rational quadratic curve; the weight parameter $w = 2.2$

As can be seen from *Figure* 2.5 the point $\mathbf{s} = \mathbf{c}(\frac{1}{2})$ of a conic segment in standard form of equation (2.3) is called the *shoulder point.* It can be computed either from equation (2.3) as

$$\mathbf{s} = \mathbf{c}(0.5)$$

or from

$$\mathbf{s} = \frac{1}{2}\mathbf{q}_0 + \frac{1}{2}\mathbf{q}_1, \tag{2.5}$$

where

$$\mathbf{q}_0 = \frac{\mathbf{b}_0 + w\mathbf{b}_1}{1+w}, \mathbf{q}_1 = \frac{w\mathbf{b}_1 + \mathbf{b}_2}{1+w} \tag{2.6}$$

are called the *characteristic points* as seen from the *Figure* 2.5. The *shoulder tangent* is spanned by $\mathbf{q}_0$ and $\mathbf{q}_1$.

Note that the shoulder tangent is parallel to $[\mathbf{b}_0\ ,\ \mathbf{b}_2]$; check the *Figure* 2.6. As a consequence, the weight

$$w = \frac{||\mathbf{s} - \mathbf{m}||}{||\mathbf{b}_1 - \mathbf{s}||} \tag{2.7}$$

where

$$||\mathbf{s} - \mathbf{m}|| = \sqrt{(s_x - m_x)^2 + (s_y - m_y)^2},$$

$$||\mathbf{b}_1 - \mathbf{s}|| = \sqrt{(b_{1x} - s_x)^2 + (b_{1y} - s_y)^2}$$

where $\mathbf{m}$ is the mid-point of $\mathbf{b}_0$ and $\mathbf{b}_2$ and $\mathbf{s} = (s_x, s_y)$; $\mathbf{m} = (m_x, m_y)$; $\mathbf{b}_1 = (b_{1x}, b_{1y})$.

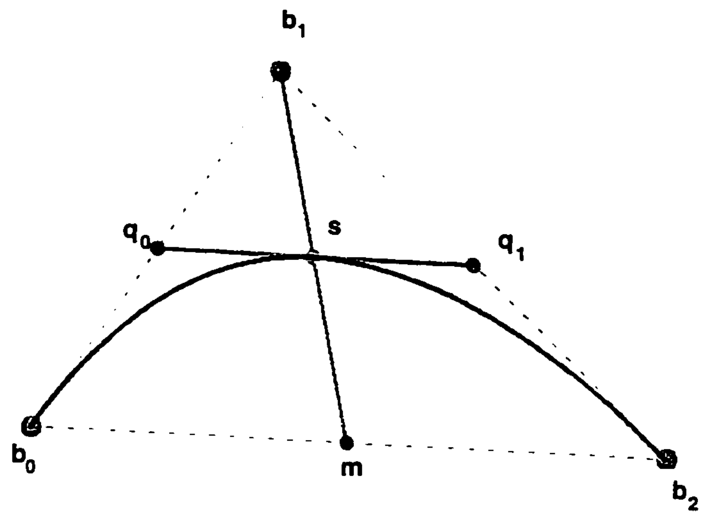


Figure 2.6: A rational quadratic curve; the weight parameter given $w = 1.0$ and the weight parameter found is also 1.0

As can be seen from the *Figure* 2.6, the conic drawn with a given weight w and conic drawn by finding the weight from the above equation (2.7) are same.

If two conics possess the same polygon but different weights, say, w and $\hat{w}$, the difference $\mathbf{d}$ between their respective shoulder points is given by

$$\mathbf{d} = \frac{w - \hat{w}}{(1+w)(1+\hat{w})}[\mathbf{b}_1 - \mathbf{m}].$$

The *curvature* κ of c at the endpoints is given by

$$\kappa(0) = \frac{\tau}{w^2\rho^3}, \qquad \kappa(1) = \frac{\tau}{w^2\lambda^3} \tag{2.8}$$

where τ denotes the area of the triangle formed by the control polygon ; that is

$$\tau = \frac{1}{2}det(\mathbf{b}_1 - \mathbf{b}_0, \mathbf{b}_2 - \mathbf{b}_1) = \frac{1}{2}((b_{1x} - b_{0x})(b_{2y} - b_{1y}) - (b_{2x} - b_{1x})(b_{1y} - b_{0y}))$$

and

$$\rho = ||\mathbf{b}_1 - \mathbf{b}_0|| = \sqrt{(b_{1x} - b_{0x})^2 + (b_{1y} - b_{0y})^2},$$

$$\lambda = ||\mathbf{b}_2 - \mathbf{b}_1|| = \sqrt{(b_{2x} - b_{1x})^2 + (b_{2y} - b_{1y})^2}$$

where $\mathbf{b}_0 = (b_{0x}, b_{0y})$, $\mathbf{b}_1 = (b_{1x}, b_{1y})$, $\mathbf{b}_2 = (b_{2x}, b_{2y})$ are the control points.

Note that κ denotes the signed curvature, since τ may be positive or negative. It is found that for weight $w < 1$ and any set of control points the curvature at the start $\kappa(0)$ is more than the curvature at the end $\kappa(1)$; for weight $w > 1$ and any set of control points the curvature at the start $\kappa(0)$ is less than the curvature at the end $\kappa(1)$.

Though the standard form is used most often there are some disadvantages of this form, like, here it is difficult to answer if a point x lies on the conic and also all the parameters (the control polygon, shape parameters) should be given to get a conic. In some cases it may be the situation that only the control polygon and a point on the conic is known but the shape parameter is unknown, then it is better to use other form to get the unknown shape parameter. All these shortcomings of this form can be overcome by other forms as described next.

2.3 Implicit form

Consider a triangle with vertices $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ and a fourth point $\mathbf{p}$, all in $\mathbf{E}^2$ as in [28]. It is always possible to write $\mathbf{p}$ as a barycentric combination of $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$:

$$\mathbf{p} = u\mathbf{a} + v\mathbf{b} + w\mathbf{c}$$

In order for the above equation to be a barycentric combination (and hence to be geometrically meaningful), it is required that

$$u + v + w = 1$$

The coefficients $\mathbf{u} := (\mathrm{u, v, w})$ are called *barycentric coordinates* of $\mathbf{p}$ with respect to $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$. Thus, if the four points $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ and $\mathbf{p}$ are given, we can always determine

$\mathbf{p}$'s barycentric coordinates u, v, w by an application of Cramer's rule:

$$u = \frac{area(\mathbf{p}, \mathbf{b}, \mathbf{c})}{area(\mathbf{a}, \mathbf{b}, \mathbf{c})}, v = \frac{area(\mathbf{a}, \mathbf{p}, \mathbf{c})}{area(\mathbf{a}, \mathbf{b}, \mathbf{c})}, w = \frac{area(\mathbf{a}, \mathbf{b}, \mathbf{p})}{area(\mathbf{a}, \mathbf{b}, \mathbf{c})}. \tag{2.9}$$

where

$$area(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \begin{vmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ 1 & 1 & 1 \end{vmatrix}$$

$$= \frac{1}{2} \times (a_x(b_y - c_y) - b_x(a_y - c_y) + c_x(a_y - b_y))$$

It is important to note that for equation (2.9) to be well-defined we must have area($\mathbf{a}$,$\mathbf{b}$,$\mathbf{c}$) $\neq 0$, which means that **a, b, c** must not lie on a straight line.

Barycentric coordinates are affinely invariant: let $\mathbf{p}$ have barycentric coordinates u, v, w with respect to **a, b, c**. Now map all four points to another set of four points by an affine map Φ. Then Φ has the same barycentric coordinates u, v, w with respect to $\Phi\mathbf{a}$, $\Phi\mathbf{b}$, $\Phi\mathbf{c}$.

Every conic $\mathbf{c}(t)$ has an *implicit representation* of the form

$$f(x, y) = 0$$

In order to find this representation, recall that $\mathbf{c}(t)$ may be written in terms of barycentric coordinates of the polygon vertices $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$:

$$\mathbf{c}(t) = \tau_0\mathbf{b}_0 + \tau_1\mathbf{b}_1 + \tau_2\mathbf{b}_2; \tag{2.10}$$

as explained in the previous discussion of barycentric coordinates. Since $\mathbf{c}(t)$ may also be written as a rational Bezier curve of equation (2.1), and since both representations are unique, we may compare the coefficients of the $\mathbf{b}_i$:

$$\tau_0 = \frac{[w_0(1-t)^2]}{D}, \tag{2.11}$$

$$\tau_1 = \frac{[2w_1 t(1-t)]}{D}, \tag{2.12}$$

$$\tau_2 = \frac{[w_2 t^2]}{D} \tag{2.13}$$

where $D = \sum w_i B_i^2$. We may solve equation (2.11) and (2.13) for $(1-t)$ and t, respectively. Inserting both expressions into equation (2.12) yields

$$\tau_1^2 = 4\frac{\tau_0 \tau_2 w_1^2}{w_0 w_2}.$$

which can be written more symmetrically as

$$\frac{\tau_1^2}{\tau_0 \tau_2} = 4\frac{w_1^2}{w_0 w_2}.$$

Conic from two points and tangents plus another point: Now, using the previously explained theory about barycentric coordinates the implicit form of a rational quadratic in its standard form where $w_0 = w_2 = 1$ is given by

$$\tau_1^2 = 4w^2 \tau_0 \tau_2, \tag{2.14}$$

where the τ_i are barycentric coordinates of a point $\mathbf{p}$ on the rational quadratic with respect to the triangle $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ such that

$$\tau_0 = \frac{area(\mathbf{p}, \mathbf{b}_1, \mathbf{b}_2)}{area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2)}$$

$$\tau_1 = \frac{area(\mathbf{b}_0, \mathbf{p}, \mathbf{b}_2)}{area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2)}$$

$$\tau_2 = \frac{area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{p})}{area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2)}$$

where

$$area(\mathbf{p}, \mathbf{b}_1, \mathbf{b}_2) = (p_x(b_{1y} - b_{2y}) - b_{1x}(p_y - b_{2y}) + b_{2x}(p_y - b_{1y}))$$

$$area(\mathbf{b}_0, \mathbf{p}, \mathbf{b}_2) = (b_{0x}(p_y - b_{2y}) - p_x(b_{0y} - b_{2y}) + b_{2x}(b_{0y} - p_y))$$

$$area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{p}) = (b_{0x}(b_{1y} - p_y) - b_{1x}(b_{0y} - p_y) + p_x(b_{0y} - b_{1y}))$$

$$area(\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2) = (b_{0x}(b_{1y} - b_{2y}) - b_{1x}(b_{0y} - b_{2y}) + b_{2x}(b_{0y} - b_{1y}))$$

Thus, given a control polygon and an arbitrary point on the conic, we may find the weight w by finding the barycentric coordinates of the point and then solving the equation (2.14) for w. This is one of the important application of this implicit form which was not possible by the standard form. As seen from the *Figure* 2.7 the arbitrary selected point is taken such that it lies within the control polygon and it passes thru the curve. Also a point at $t = 0.4$ on the conic is drawn by taking the weight $w = 2.2$ and using that point the weight is found by using barycentric coordinates (it is found to be also 2.2) and the curve is drawn and as seen it coincides with the previous one.

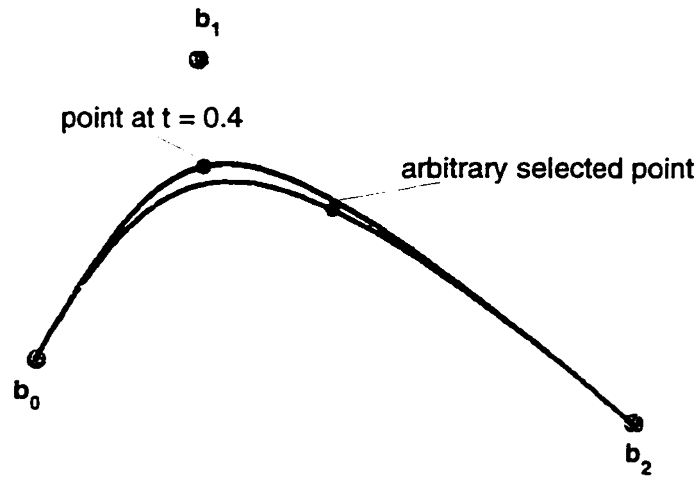


Figure 2.7: A rational quadratic drawn by *implicit representation* using barycentric coordinates

The implicit form has one more important application: Suppose we are given a conic section $\mathbf{c}$ and an arbitrary point $\mathbf{x} \in \mathbf{E}^2$. Does $\mathbf{x}$ lie on $\mathbf{c}$? This can be hard to answer if $\mathbf{c}$ is given in parametric form of equation (2.1). However, using the implicit form, this question is easily answered. First compute the barycentric coordinates τ_0, τ_1, τ_2 of $\mathbf{x}$ with respect to $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2$. Then insert τ_0, τ_1, τ_2 into equation (2.14). If equation (2.14) is satisfied $\mathbf{x}$ lies on the conic.

2.4 Recursive algorithmic form

We may evaluate equation (2.1) by evaluating numerator and denominator separately and then dividing through. A rational quadratic may be evaluated by a *recursive algorithm* as in [28].

Define

$$\mathbf{b}_i^r(t) = (1-t)\frac{w_i^{r-1}}{w_i^r} b_i^{r-1} + t\frac{w_{i+1}^{r-1}}{w_i^r} b_{i+1}^{r-1}. \tag{2.15}$$

where

$$w_i^r(t) = (1-t)w_i^{r-1}(t) + t w_{i+1}^{r-1}(t); \tag{2.16}$$

and $r = 1, 2$; $i = 0, \ldots, 2-r$; $w_i^0(t) = w_i$; $b_i^0(t) = b_i$.

Then $\mathbf{c}(t) = \mathbf{b}_0^2(t)$. Using the equation (2.15) the rational quadratic can be drawn as explained in the following. For each value of the parameter $t, 0 \leq t \leq 1$, the weights w_0^1, w_1^1, w_0^2 are found, then the intermediate Bezier points $\mathbf{b}_0^1, \mathbf{b}_1^1$ are found

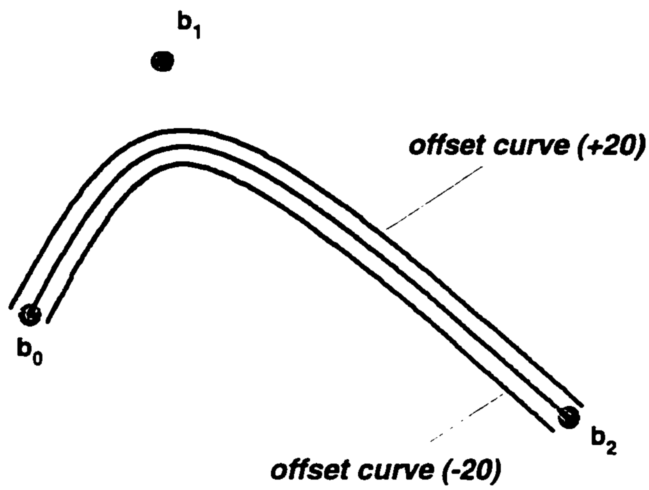


Figure 2.8: A rational quadratic drawn using *recursive algorithm*; the weight parameter $w = 0.4$

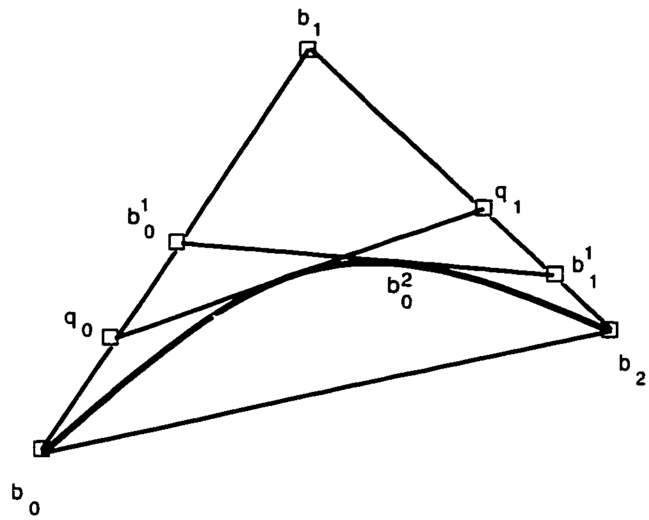


Figure 2.9: A rational quadratic drawn with *recursive algorithm*

and then finally all these values are substituted in $\mathbf{b}_0^2(t)$ to get a point on the conic for that parameter value t as seen in the *Figure* 2.9. As the parameter t varies from 0 to 1, the points obtained from $\mathbf{b}_0^2(t)$ forms the required conic while the intermediate Bezier point $\mathbf{b}_0^1$ forms a straight line between the control points $\mathbf{b}_0$ and $\mathbf{b}_1$, while the intermediate Bezier point $\mathbf{b}_1^1$ forms a straight line between the control points $\mathbf{b}_1$ and $\mathbf{b}_2$. Also for a particular value of t the line between $\mathbf{b}_0$ and $\mathbf{b}_1$ meets the conic at $\mathbf{b}_0^2(t)$ and forms the tangent at $\mathbf{b}_0^2(t)$ to the conic as seen in the *Figure* 2.9.

The equations (2.15) and (2.16) can be written more explicitily as:

$$w_0^1(t) = (1-t)w_0^0(t) + tw_1^0(t) = (1-t) + wt$$

$$w_1^1(t) = (1-t)w_1^0(t) + tw_2^0(t) = (1-t)w + t$$

$$w_0^2(t) = (1-t)w_0^1(t) + tw_1^1(t) = (1-t)^2 + 2wt(1-t) + t^2$$

$$\mathbf{b}_0^1(t) = (1-t)\frac{w_0^0}{w_0^1}\mathbf{b}_0^0 + t\frac{w_1^0}{w_0^1}\mathbf{b}_1^0$$

$$\mathbf{b}_0^1(t) = (1-t)\frac{1}{(1-t)+wt}\mathbf{b}_0 + t\frac{w}{(1-t)+wt}\mathbf{b}_1 \tag{2.17}$$

$$\mathbf{b}_1^1(t) = (1-t)\frac{w_1^0}{w_1^1}\mathbf{b}_1^0 + t\frac{w_2^0}{w_1^1}\mathbf{b}_2^0$$

$$\mathbf{b}_1^1(t) = (1-t)\frac{w}{(1-t)w+t}\mathbf{b}_1 + t\frac{1}{(1-t)w+t}\mathbf{b}_2 \tag{2.18}$$

$$\mathbf{b}_0^2(t) = (1-t)\frac{w_0^1}{w_0^2}\mathbf{b}_0^1 + t\frac{w_1^1}{w_0^2}\mathbf{b}_1^1$$

$$\mathbf{b}_0^2(t) = (1-t)\frac{(1-t)+wt}{(1-t)^2+2wt(1-t)+t^2}\mathbf{b}_0^1 + t\frac{(1-t)w+t}{(1-t)^2+2wt(1-t)+t^2}\mathbf{b}_1^1$$

Four collinear points - a, b, c and d define the so-called *cross ratio*:

$$cr(a,b,c,d) = ratio(a,b,d)/ratio(a,c,d)$$

where

$$ratio(a,b,c) = ||b-a||/||c-b||$$

With the points $\mathbf{q}_i$ as

$$\mathbf{q}_i^r(t) = \frac{w_i^r\mathbf{b}_i^r + w_{i+1}^r\mathbf{b}_{i+1}^r}{w_i^r + w_{i+1}^r}$$

one has

$$cr(b_0, q_0, \mathbf{b}_0^1, \mathbf{b}_1) = cr(\mathbf{b}_1, q_1, \mathbf{b}_1^1, b_2) = cr(0, 1/2, t, 1) = \frac{(1-t)}{t} \tag{2.19}$$

While computationally more involved than the straightforward algebraic approach, this generalized *de Casteljau algorithm* has the advantage of being numerically stable; it uses only convex combinations, provided the weights are positive and $t \in [0,1]$.

This recursive form is very useful when the conic has to be subdivided to improve its smoothness and also while drawing its offsets.

2.5 Subdivision of Conic

The intermediate Bezier points $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ of the previous recursive construction may be used to *subdivide* the curve at parameter value t. As seen from the *Figure*

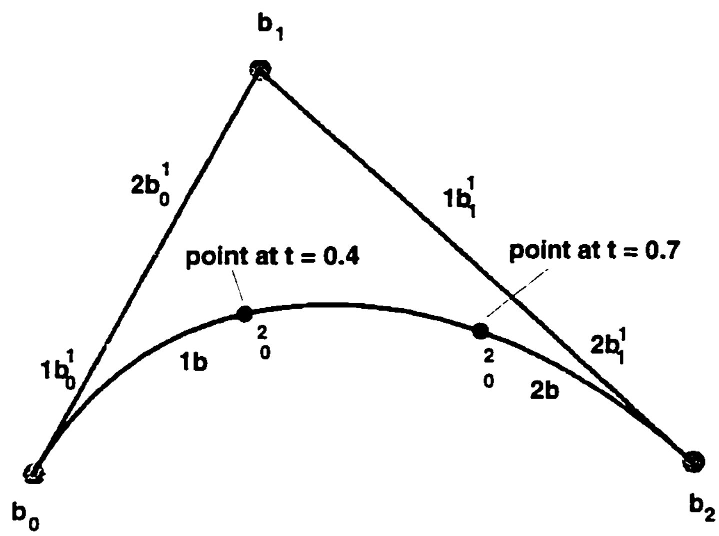


Figure 2.10: A rational quadratic subdivided at two places: at $t = 0.4$ and at $t = 0.7$ parameter values; the weight parameter $w = 0.7$

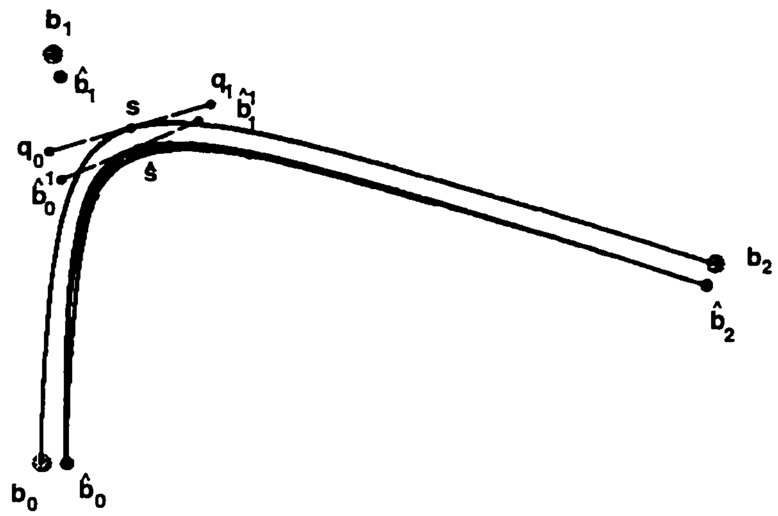


Figure 2.11: A rational quadratic and its offsets; the offsets are drawn using the subdivide and standardize procedure; the conic is subdivided at $t = \frac{1}{2}$; the weight parameter $w = 3.2$

2.10, when the curve is subdivided at a point when the parameter value $t = 0.4$: the first left segment has Bezier points $\mathbf{b_0}, \mathbf{1b_0^1}, \mathbf{1b_0^2}$ with weights $1, w_0^1(t), w_0^2(t)$; the second right segment has Bezier points $\mathbf{1b_0^2}, \mathbf{1b_1^1}, \mathbf{b_2}$ with weights $w_0^2(t), w_1^1(t), 1$ and when the curve is subdivided at a point when the parameter value $t = 0.7$: the first left segment has Bezier points $\mathbf{b_0}, \mathbf{2b_0^1}, \mathbf{2b_0^2}$ with weights $1, w_0^1(t), w_0^2(t)$; the second right segment has Bezier points $\mathbf{2b_0^2}, \mathbf{2b_1^1}, \mathbf{b_2}$ with weights $w_0^2(t), w_1^1(t), 1$. *As seen all the original and subdivided curves coincide.* In general the weight w_0^2 does not equal 1 and it may be changed by the process of standardization as explained in the previous section.

In order to standardize the two segments resulting from the subdivision as in [28] the following substitutions can be made

$$w_0^1 \leftarrow \frac{w_0^1}{\sqrt{w_0^2}}, w_0^2 \leftarrow 1, w_1^1 \leftarrow \frac{w_1^1}{\sqrt{w_0^2}}.$$

Now both segments are again of the standard form of equation (2.3). In the special case where $t = \frac{1}{2}$, we get $w_0^1 = w_1^1 = w_0^2 = \frac{w+1}{2}$. Thus, w_0^1, w_1^1 have to be simply substituted by their square roots as seen in *Figures* 2.10 and 2.11.

2.6 Curvature continuous piecewise rational quadratic

In the previous section we were discussing about a single segment of conic. We now move ahead and consider planar curves that consist of n rational quadratic segments (or, equivalently, of n conic segments) as in [28]. The ith segment is defined by the control polygon $\mathbf{b}_{2i}, \mathbf{b}_{2i+1}, \mathbf{b}_{2i+2}; i = 0, ..., n-1$ and the weight w_{2i+1} - called an *inner weight* - that is associated with $\mathbf{b}_{2i+1}$. The points $\mathbf{b}_{2i}$ that are common to two adjacent segments are called *junction points.*

The $(i-1)$st and ith curve segments are *tangent continuous* at $\mathbf{b}_{2i}$ if $\mathbf{b}_{2i-1}, \mathbf{b}_{2i}, \mathbf{b}_{2i+1}$ are collinear and distinct such that $\mathbf{b}_{2i}$ is located between $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$.

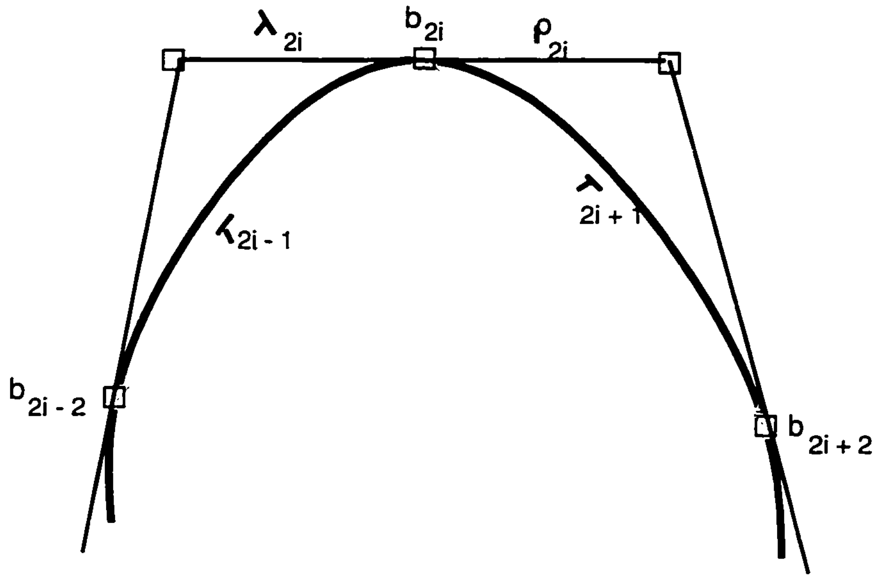


Figure 2.12: Curvature continuity: two adjacent rational quadratic segments

They are *curvature continuous* at $\mathbf{b}_{2i}$ if in addition the curvature at the end of the $(i-1)$st segment is equal to the curvature at the start of the ith segment. From equation (2.2) the curvature at the end point $\mathbf{b}_{2i}$ of the $(i-1)$st segment is given as

$$\frac{\tau_{2i-1}}{w_{2i-1}^2 \lambda_{2i}^3}; i = 1, ..., n-1. \tag{2.20}$$

where

$$\tau_{2i-1} = det(\mathbf{b}_{2i-2}, \mathbf{b}_{2i-1}, \mathbf{b}_2 i), \lambda_{2i} = ||\mathbf{b}_{2i} - \mathbf{b}_{2i-1}||$$

and $\mathbf{b}_{2i-2}, \mathbf{b}_{2i-1}, \mathbf{b}_{2i}$ are the control points of $(i-1)$st segment.

From equation (2.2) the curvature at the starting point b_{2i} of the ith segment is given as

$$\frac{\tau_{2i+1}}{w_{2i+1}^2 \rho_{2i}^3}; i = 1, ..., n-1 \tag{2.21}$$

where

$$\tau_{2i+1} = det(\mathbf{b}_{2i}, \mathbf{b}_{2i+1}, \mathbf{b}_{2i+2}), \rho_{2i} = ||\mathbf{b}_{2i+1} - \mathbf{b}_2 i||$$

and $\mathbf{b}_{2i}, \mathbf{b}_{2i+1}, \mathbf{b}_{2i+2}$ are the control points of ith segment, as seen in the *Figure* 2.12. Now to obtain the curvature continuity between the two $(i-1)$st and ith segments, the equations (2.20) and (2.21) should be equated to get

$$\frac{\tau_{2i-1}}{w_{2i-1}^2 \lambda_{2i}^3} = \frac{\tau_{2i+1}}{w_{2i+1}^2 \rho_{2i}^3}$$

which can be rewritten as

$$w_{2i+1}^2 = w_{2i-1}^2 \frac{\lambda_{2i}^3 \tau_{2i+1}}{\rho_{2i}^3 \tau_{2i-1}} \tag{2.22}$$

The above equation (2.22) specifies that given the weight of the previous segment (segment $(i-1)$), the weight of the next segment (segment i), can be found. The equation (2.22) can only be computed if the right hand side of equation is nonnegative. Since the λ_{2i} and ρ_{2i} are positive by definition, the condition for positivity of the right-hand side is that τ_{2i+1} and τ_{2i-1} be of the same sign which geometrically means that $\mathbf{b}_{2i-2}$ and $\mathbf{b}_{2i+2}$ must lie on the same side of the straight line segment $[\mathbf{b}_{2i-1}, \mathbf{b}_{2i+1}]$.

Thus after assigning an arbitrary first weight w_1 for the first segment, the remaining weights can be computed sequentially from equation (2.22). Using the equation (2.22) we can now consider the problem of getting a curvature continuous piecewise rational quadratic curve as in [28].

Thus when the points $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_3, ..., \mathbf{b}_{2i+1}, ..., \mathbf{b}_{2n-1}, \mathbf{b}_{2n}$, forming a locally convex polygon are given, we have to find the junction points $\mathbf{b}_{2i}$ and inner weights w_{2i+1} such that the resulting piecewise rational quadratic curve is curvature continuous.

For a solution, the junction points $\mathbf{b}_{2i}$ can be placed anywhere between the $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$. First, we will consider the case where the $\mathbf{b}_{2i}$ are placed at the mid-point between $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$. After finding the junction points $\mathbf{b}_{2i}$ (here, at the mid-point between $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$), the initial weight w_1 of the first segment is substituted in equation (2.22) to get the weight for the second segment w_2, then this w_2 is substituted in equation (2.22) to get w_3 and so on to get the weights of all the n segments and thus a *curvature continuous piecewise approximatory rational quadratic* is obtained.

The other weights obtained from the initial weight w_1 tries to be close to the weight w_1, but as the w_1 increases more and more the other weights differs from w_1 greatly as seen from the *Figures* 2.13, 2.14, 2.15, 2.16, 2.17 where four segments of conics are joined in such a way that a curvature continuous rational quadratic curve where the junction points $\mathbf{b}_{2i}$ are at mid-points and the curve passes through those junction points and does not interpolate the given points $\mathbf{b}_1, \mathbf{b}_3, ..., \mathbf{b}_{2i+1}$. Also as

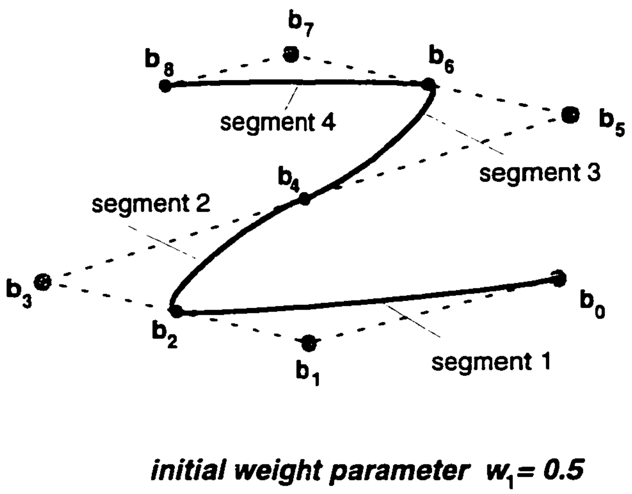


Figure 2.13: four segments of curvature continuous piecewise rational quadratic with junction points $\mathbf{b}_2, \mathbf{b}_4, \mathbf{b}_6, \mathbf{b}_8$ at mid-points between the given control points $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_3, \mathbf{b}_5, \mathbf{b}_7$

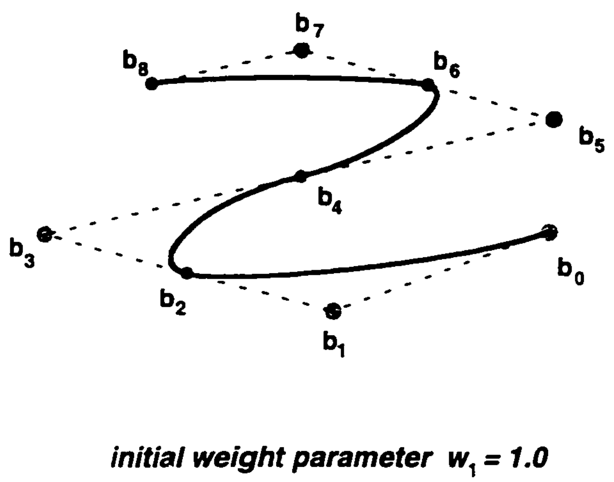


Figure 2.14: four segments of curvature continuous piecewise rational quadratic with junction points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ at mid-points between the given control points $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$

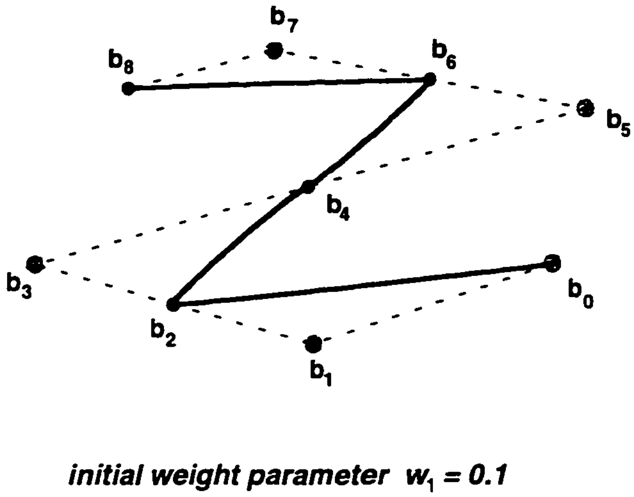


Figure 2.15: four segments of curvature continuous piecewise rational quadratic with junction points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ at mid-points between the given control points $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$

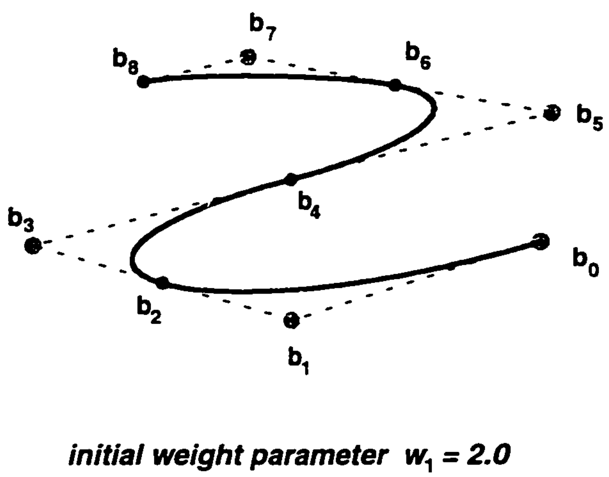


Figure 2.16: four segments of curvature continuous piecewise rational quadratic with junction points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ at mid-points between the given control points $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$

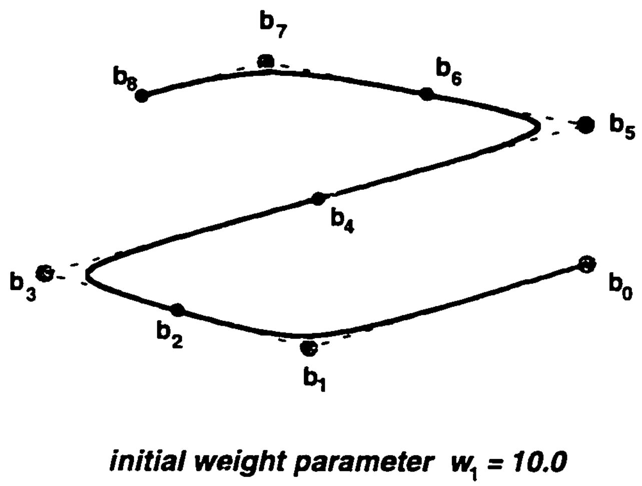
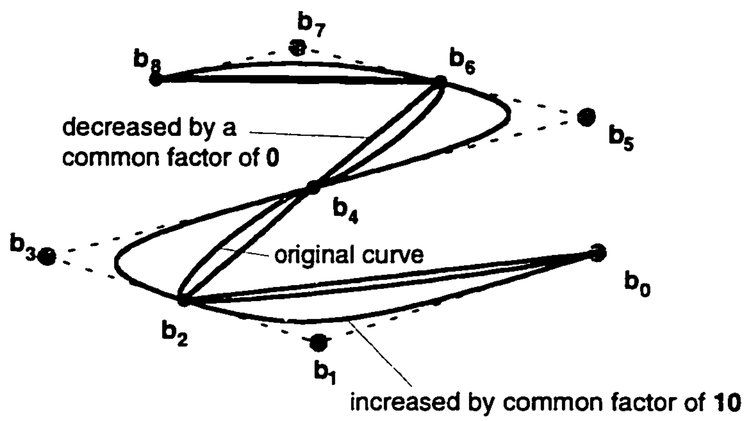


Figure 2.17: four segments of curvature continuous piecewise rational quadratic with junction points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ at mid-points between the given control points $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$

seen from the *Figures* as the weight w_1 and weights of other segments found increases the curve tends to draw closer to the control polygons and finally coincides with the control polygon when w becomes higher and higher.

Suppose now that a piecewise rational quadratic curve has continuous curvature at all junction points. If we change all inner weights w_{2i+1} by a common factor, equation (2.22) tells us that we still have a continuous curvature at the junction points. However, the shape of the curve has changed: If we increase the common factor (that is, if all the inner weights are multiplied by a common factor), the curve segments will tend toward their control polygons; if we decrease the common factor (that is, if all the inner weights are divided by a common factor), the curve segments will tend toward the chords $[\mathbf{b}_{2i}, \mathbf{b}_{2i+2}]$ as seen from the *Figures* 2.18, 2.19, 2.20. As seen from the *Figure* 2.18 when the common factor is decreased by 0, the weight parameter w becomes 0 and the curve forms a straight line. As seen from the *Figure* 2.19 when the common factor is decreased by 10, the weight parameter w comes comes very near to 0 and the curve almost forms a straight line. This common factor may therefore be termed as a *shape parameter*. Assigning a larger or smaller value to the first weight w_1 has the same effect as seen from *Figure* 2.21.

Now consider the problem where a piecewise rational quadratic with its inner weights $w_{2i+1} = 1$ for all i. Such a curve is also called a quadratic B-spline curve and is in general not curvature continuous. To make this curve a curvature continuous curve with the same control polygons new weights w_{2i+1} have to be found from



initial weight parameter $w_1 = 0.5$

Figure 2.18: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of applying common factor; the junction points are at mid-points

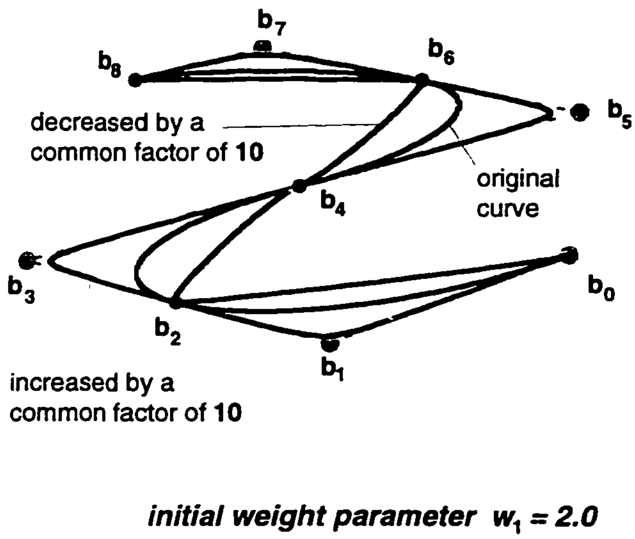


Figure 2.19: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of applying common factor; the junction points are at mid-points

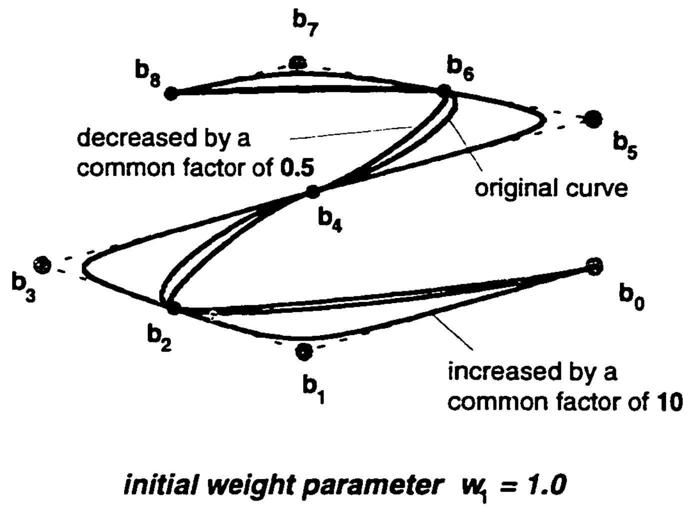


Figure 2.20: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of applying common factor; the junction points are at mid-points

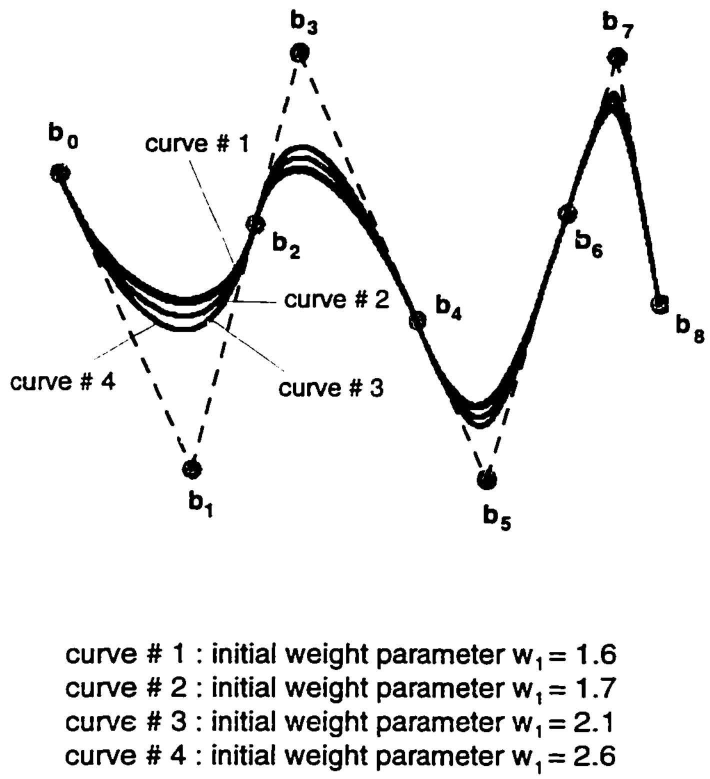


Figure 2.21: three segments of curvature continuous piecewise rational quadratic is drawn to check the effect of changing the initial weight; the junction points are **not** at mid-points

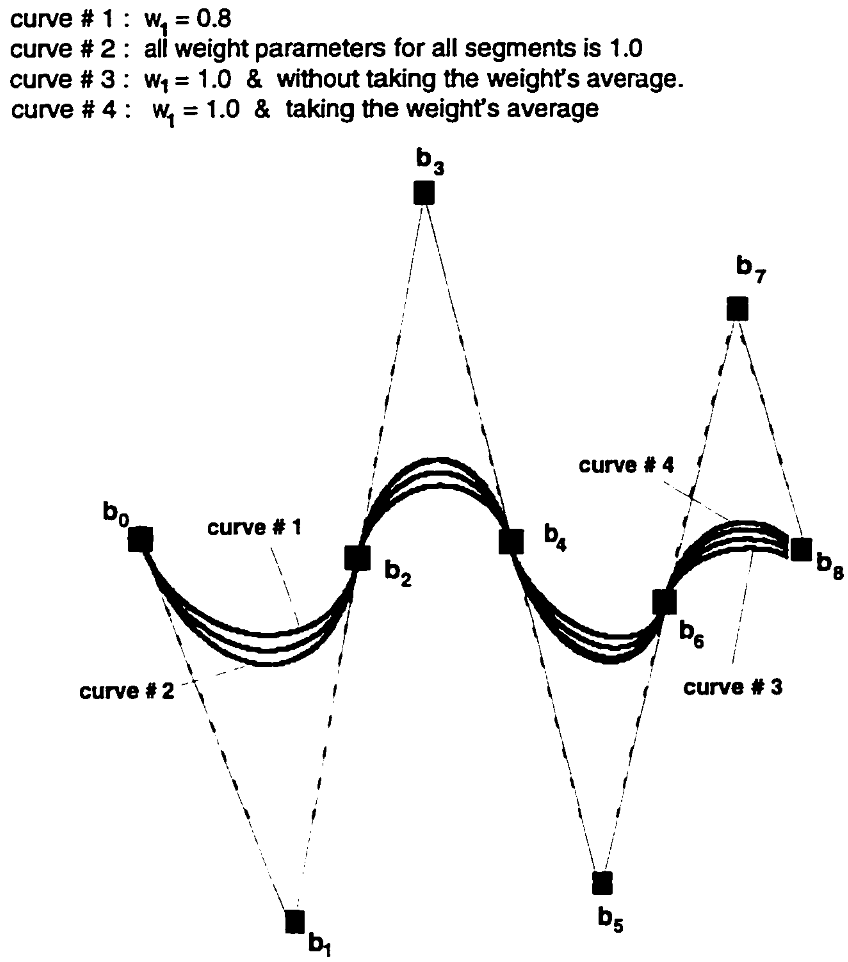


Figure 2.22: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of taking the weights as average; the junction points are at mid-points

equation (2.22) as seen from the *Figure* 2.22.

Now consider another case where it is required that the w_{2i+1} deviate from unity as little as possible. To achieve this, after computing all weights starting with $w_1 = 1$, we may correct each inner weight by dividing by the average of all inner weights. As seen from the *Figure* 2.22 the curve drawn with $w_{2i+1} = 1$ for all is converted into curvature continuous curve and then all the inner weights are divided by average to get back the curve which is smaller to the curve where all $w_{2i+1} = 1$ but now the curve is curvature continuous.

Another interesting case would be to determine which of the two averages would be better: the arithmetic mean or the geometric mean. See *Figure* 2.23 where the initial weight is taken as 1.0 and then two curves are drawn by taking the arithmetic mean and geometric mean respectively of the weights and as seen both the curves are almost coinciding and also the curve where the weights of all segments are taken as 1.0.

Till now we have been considering the junction points to be at the mid-point. But now we'll consider the junction points to be at some other place. To be specific we'll consider the junction points $\mathbf{b}_{2i}$ to lie nearer the points $\mathbf{b}_{2i+1}$ and be collinear with $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$. The junction points $\mathbf{b}_{2i}$ lies between $\mathbf{b}_{2i-1}$ and $\mathbf{b}_{2i+1}$ such that its distance from these two points is kept at the ratio of 3 : 1 respectively. As seen from the *Figures* 2.24, 2.21 when the junction points are not at the mid-point the curves tend to draw closer to the control polygons with each next segment.

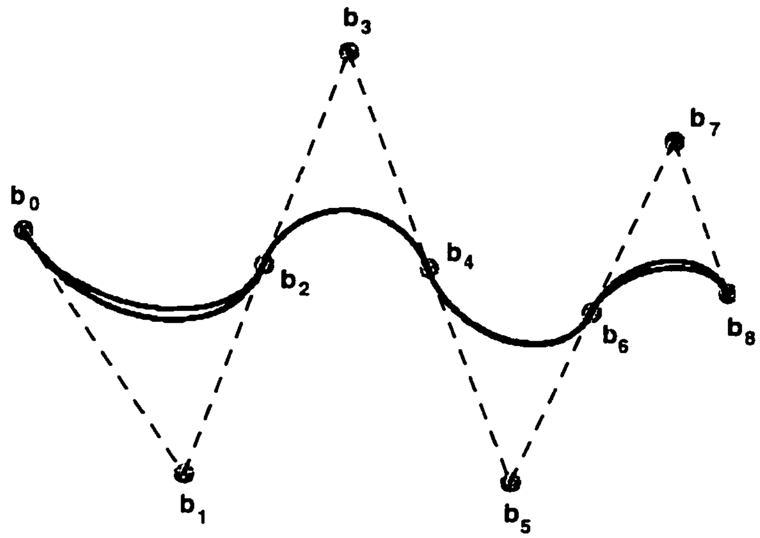


Figure 2.23: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of taking the weights as average; the junction points are at mid-points

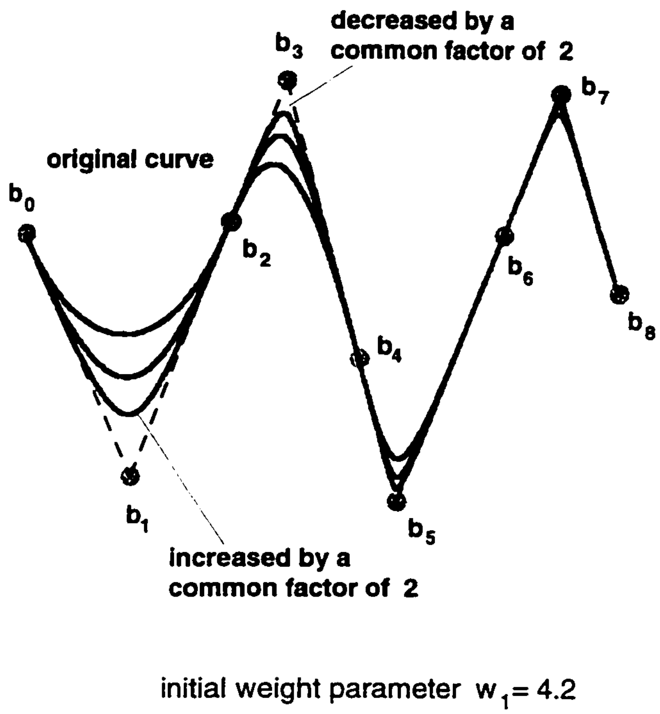


Figure 2.24: four segments of curvature continuous piecewise rational quadratic is drawn to check the effect of applying common factor to weight parameter; the junction points are **not** at mid-points

The *Figures* 2.24, 2.21 also proves that the curvature continuity at all junction points can be maintained though the shape of the curve gets changed by either applying a common factor to all inner weights or by assigning a larger or smaller value to the initial weight w_1.

Thus, from all the previous cases considered so far, it can be concluded that the control polygon provides a rough curve shape and determining the junction points and weights amounts to fine-tune of the curve shape.

Conic from two points and tangents plus a third tangent: Suppose, we are given two points $\mathbf{b}_0$ and $\mathbf{b}_2$ of a conic section, together with the corresponding tangents, which meet at $\mathbf{b}_1$. Instead of giving the weight w, consider we are given a third tangent passing through two points $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ and the weight w is to be found so that the conic can be defined.

This can be achieved by the application of the recursive algorithm. The characteristic points q_0 and q_1 are given as

$$\mathbf{q}_0 = \frac{\mathbf{b}_0 + w\mathbf{b}_1}{1+w}, \mathbf{q}_1 = \frac{w\mathbf{b}_1 + \mathbf{b}_2}{1+w}$$

Thus finding the weight w is equivalent to finding $\mathbf{q}_0$ and $\mathbf{q}_1$.

Consider the *cross-ratio* of four collinear points $\mathbf{b}_0, \mathbf{q}_0, \mathbf{b}_0^1, \mathbf{b}_1$

$$cr(b_0, q_0, b_0^1, b_1) = \frac{ratio(b_0, q_0, b_1)}{ratio(b_0, b_0^1, b_1)} \tag{2.23}$$

where

$$ratio(b_0, q_0, b_1) = \frac{||q_0 - b_0||}{||b_1 - q_0||}, ratio(b_0, b_0^1, b_1) = \frac{||b_0^1 - b_0||}{||b_1 - b_0^1||}$$

Now,

$$q_0 - b_0 = \frac{b_0 + wb_1}{1+w} - b_0 = \frac{b_0 + wb_1 - b_0 - wb_0}{1+w} = \frac{w(b_1 - b_0)}{1+w}$$

$$b_1 - q_0 = b_1 - \frac{b_0 + wb_1}{1+w} = \frac{b_1 + wb_1 - b_0 - wb_1}{1+w} = \frac{(b_1 - b_0)}{1+w}$$

so that

$$ratio(b_0, q_0, b_1) = \frac{w(b_1 - b_0)/(1+w)}{(b_1 - b_0)/(1+w)} = w \tag{2.24}$$

Now, consider the cross-ratio of the four collinear points $\mathbf{b}_1, \mathbf{q}_1, \mathbf{b}_1^1, \mathbf{b}_2$

$$cr(b_1, q_1, b_1^1, b_2) = \frac{ratio(b_1, q_1, b_2)}{ratio(b_1, b_1^1, b_2)} \tag{2.25}$$

where

$$ratio(b_1, q_1, b_2) = \frac{||q_1 - b_1||}{||b_2 - q_1||}, ratio(b_1, b_1^1, b_2) = \frac{||b_1^1 - b_1||}{||b_2 - b_1^1||}$$

Now,

$$q_1 - b_1 = \frac{b_2 + wb_1}{1+w} - b_1 = \frac{b_2 + wb_1 - b_1 - wb_1}{1+w} = \frac{(b_2 - b_1)}{1+w}$$

$$b_2 - q_1 = b_2 - \frac{b_2 + wb_1}{1+w} = \frac{b_2 + wb_2 - b_2 - wb_1}{1+w} = \frac{w(b_2 - b_1)}{1+w}$$

so that

$$ratio(b_1, q_1, b_2) = \frac{(b_2 - b_1)/(1+w)}{w(b_2 - b_1)/(1+w)} = \frac{1}{w} \tag{2.26}$$

Let

$$r_0 = ratio(b_0, b_0^1, b_1), r_1 = ratio(b_1, b_1^1, b_2)$$

The above two cross-ratios equations (2.23) and (2.25) should be equal, that is,

$$cr(b_0, q_0, b_0^1, b_1) = cr(b_1, q_1, b_1^1, b_2)$$

thus from equations (2.24) and (2.26),

$$\frac{w}{r_0} = \frac{1}{wr_1}$$

or

$$\frac{r_0}{w} = r_1 w$$

and finally

$$w = \sqrt{\frac{r_0}{r_1}} \tag{2.27}$$

Thus, after finding the weight w the conic can be defined. We can also determine to which value of the parameter t the third tangent corresponds.

Using the cross-ratio property again, with equations (2.19) and (2.23), we get

$$\frac{r_0}{w} = \frac{t}{1-t}$$

from which we can determine t as

$$t = \frac{r_0}{w + r_0}$$

or equivalently, using equation (2.24) and (2.19), we get

$$\frac{1}{wr_1} = \frac{1-t}{t}$$

from which we can determine t as

$$t = \frac{wr_1}{1 + wr_1} \tag{2.28}$$

Now, after obtaining the w and t, using the previously discussed methods we can subdivide and standardize the curve.

Thus, given two points and two intersecting tangents of a segment of a conic, together with a third tangent, we have determined **(1)** the weight w, **(2)** the parameter value t corresponding to the tangent, **(3)** the corresponding point on the curve, and **(4)** the subdivision of the conic arc into two standardized subsegments.

2.7 Offsets

Offset curves are frequently used in design (e.g., font design). If $\mathbf{c}$ is given curve, then

$$\hat{\mathbf{c}}(t) = \mathbf{c}(t) + d\mathbf{n}(t)$$

defines an offset curve where d is the offset distance and $\mathbf{n}(\mathrm{t})$ is the normal of $\mathbf{c}$, pointing in the desired offset direction as done in [28]. *The offset of a conic is not usually a conic ; therefore we shall develop an approximation method.*

Let a conic segment in standard form be given by its control polygon $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ and the inner weight w. Now, there are two methods to draw an offset to this conic segment. **First method** is to offset each point of the original conic. **Second**

method is to offset just the control points $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ of the original conic to get new control points $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ for the offset curve and find the weight $\hat{w}$ of this offset curve by using the previously discussed problem of determining weight when the control polygon and a third tangent is given, and then construct the offset conic by using the equation (2.3) for standard form.

Consider the first method of drawing an offset to a given conic. In this method for each point (x, y) of the original conic at a particular parameter value of t, the tangent to that point (t_x, t_y) is found and then the normal to that point (n_x, n_y) in the desired direction are found by using the formulas:

$$n_x = \frac{-t_y}{\sqrt{t_x^2 + t_y^2}}, n_y = \frac{t_x}{\sqrt{t_x^2 + t_y^2}} \tag{2.29}$$

Then the offset point (o_x, o_y) for the point (x, y) is given by

$$o_x = x - dn_x, o_y = y - dn_y \tag{2.30}$$

Thus, by constructing the offsets for each point of the original curve as t varies from 0 to 1 an offset to that curve is formed as seen from the *Figure* 2.25.

In the second method of getting offsets, the offsets of the control polygon $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ of the original curve are found by finding the points on the curve for the parameter values of $t = 0, 1/2, 1$ and then finds the tangents and normals at these three points as done in the first case and then finds the offset points of $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ as

$$\hat{\mathbf{b}}_0 = \mathbf{b}_0 - dn_0, \hat{\mathbf{b}}_1 = \mathbf{b}_1 - dn_1, \hat{\mathbf{b}}_2 = \mathbf{b}_2 - dn_2$$

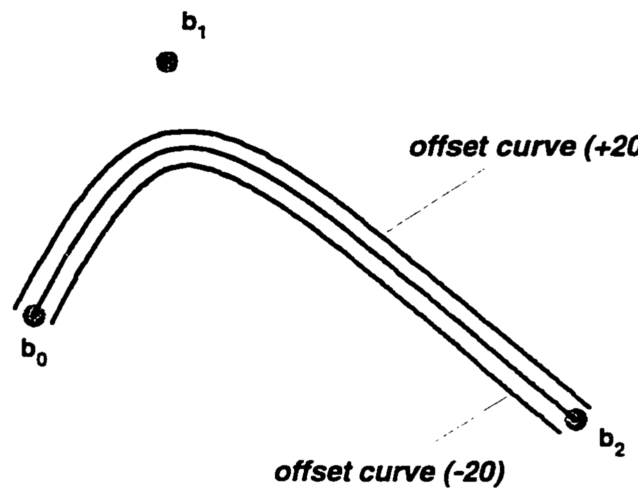


Figure 2.25: A rational quadratic with its two offsets; the weight parameter entered is 2.5

where n_0, n_1, n_2 are normals in the desired direction ar $t = 0, 1/2, 1$ that is at $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$ respectively. These offset points gives the control polygon $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ of the offset curve. Next as seen from the *Figure* 2.11 the shoulder tangent through $\mathbf{q}_0$ and $\mathbf{q}_1$ can be offset by d to obtain a tangent to the offset curve and let this offset tangent pass through $\hat{\mathbf{b}}_0^1, \hat{\mathbf{b}}_1^1, \hat{\mathbf{s}}$. Note that this $\hat{\mathbf{s}}$ is not the shoulder point of the offset approximation. The points $\mathbf{q}_0$ and $\mathbf{q}_1$ that lie on the straight line segments $[\mathbf{b}_0, \mathbf{b}_1]$ and $[\mathbf{b}_1, \mathbf{b}_2]$ respectively are basically the $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ at the parameter value of $t = 0.5$ respectively of the equations (2.17) and (2.18). To find the offset points of $\mathbf{q}_0$ and $\mathbf{q}_1$ (that is of $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$) the tangent and normal at $\mathbf{q}_0$ and $\mathbf{q}_1$ are derived as explained earlier then the offset points $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ are given by

$$\mathbf{b}_0^1 = \mathbf{b}_0^1(q_0) - d\mathbf{n}_1, \mathbf{b}_1^1 = \mathbf{b}_1^1(q_1) - d\mathbf{n}_2$$

where $\mathbf{n}_1$ and $\mathbf{n}_2$ are normals at $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ respectively.

The offset points of $\mathbf{q}_0$ and $\mathbf{q}_1$ can be found by another method. First find the offset $\hat{\mathbf{s}}$ of the shoulder point $\mathbf{s} = \mathbf{c}\frac{1}{2}$ of the original curve. Then we use the recursive algorithm procedure to draw conic by taking $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ as the control polygon and the weight w of the original curve. Now for each value of t (from 0 to 1) the point $\hat{\mathbf{c}}(t)$ thus obtained is checked for equality with the $\hat{\mathbf{s}}$, the offset of the shoulder point $\mathbf{s}$. If it is found to be equal then the corresponding $\hat{\mathbf{b}}_0^1$ and $\hat{\mathbf{b}}_1^1$ values are taken as $\mathbf{q}_0$ and $\mathbf{q}_1$.

Once we have found the control points $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ and a third tangent passing

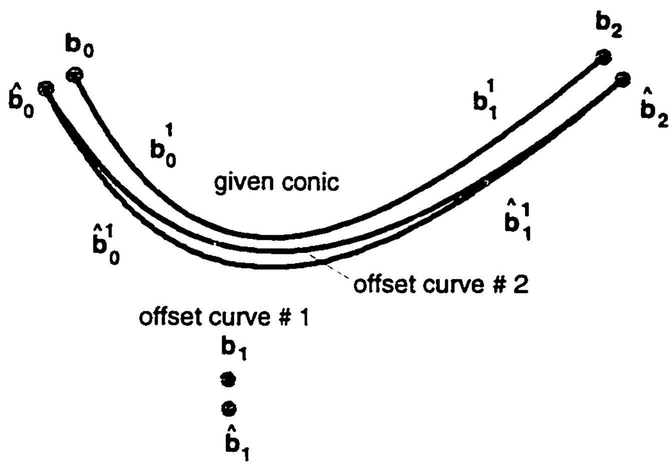


Figure 2.26: A rational quadratic and its offsets drawn by two different methods; the offset curve 1 is drawn by first method, the offset curve 2 is drawn by second method; the weight parameter $w = 1.2$

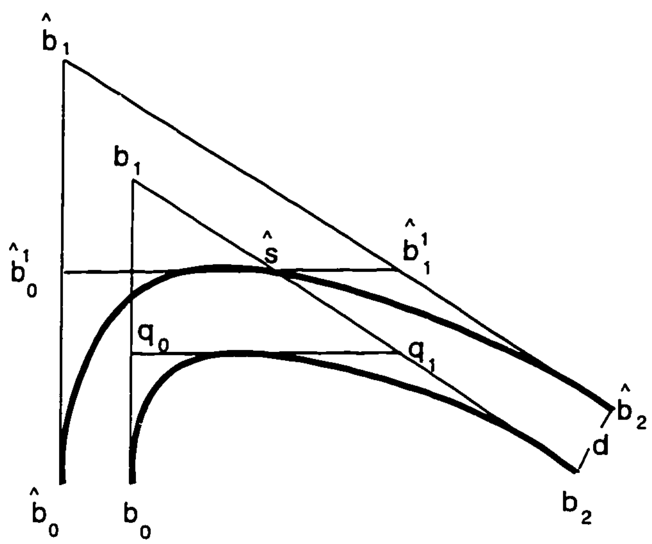


Figure 2.27: the offset of the original (bottom) curve is approximated by another conic (top curve); the offset distance is d

through $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ we can find the weight $\hat{w}$ of the offset curve by using the equation (2.27). Then finally using the equation (2.3) of standard form the offset conic approximation of the original conic curve is obtained as seen from the *Figure* 2.26.

As seen from the *Figure* 2.26 the offsets obtained by the previous two methods is almost the same. The points $\mathbf{b_0}, \mathbf{b_1} and \mathbf{b_2}$ are the given control points and the points $\hat{\mathbf{b}}_\mathbf{0}, \hat{\mathbf{b}}_\mathbf{1} and \hat{\mathbf{b}}_\mathbf{2}$ are their corresponding offsets. As shown in the *Figure* 2.27 the offset points $\hat{\mathbf{b}}_0^1$ and $\hat{\mathbf{b}}_1^1$ are found for $\mathbf{b}_0^1$ and $\mathbf{b}_1^1$ at different values of t and as seen we get different offsets which are coinciding for these different tangents passing through $\hat{\mathbf{b}}_0^1$ and $\hat{\mathbf{b}}_1^1$.

Till now the offset rational quadratic $\hat{c}$ was constructed such that the offset property was satisfied only at the parameter values $t = 0, 1/2, 1$, where t is the parameter of the original curve. In order to check the goodness of this approximation, we can do as follows: Find the points on the original curve $\mathbf{c}$ at parameter values $t = 1/4, 3/4$ to get two points $c(1/4)$ and $c(3/4)$. Now get the offset of these two points by finding the tangent and normal and using equation (2.22), thus obtaining two offset points $c_{1/4}$ and $c_{3/4}$. Now there exist a unique conic $\hat{c}_{1/4}$ with control polygon $\hat{\mathbf{b}}_0, \hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2$ that passes through $c_{1/4}$ which will be an offset to $\mathbf{c}$. Similarly there exist a unique conic $\hat{c}_{3/4}$ with control polygon $\hat{\mathbf{b}}_0, \hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2$ that passes through $c_{3/4}$ which will also be an offset to $\mathbf{c}$. Now to draw these two offsets passing through $c_{1/4}$ and $c_{3/4}$ respectively we have to compute the weights $w_{1/4}$ and $w_{3/4}$. Since we know the control polygon and a point on the conic, the weight can be found by using the

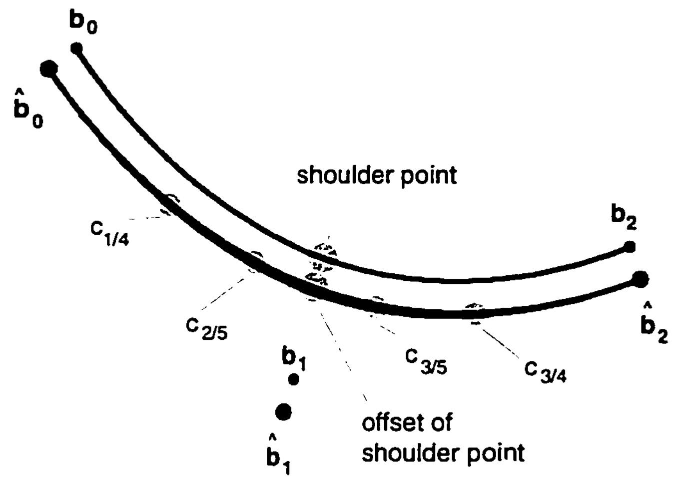


Figure 2.28: A rational quadratic and its offsets; the weight parameter $w = 0.9$

Barycentric coordinates procedure as described earlier. Now if these weights $w_{1/4}$ and $w_{3/4}$ along with the weight $\hat{w}$ are close to each other, it may be concluded that our offset approximation was satisfactory as seen from the *Figure* 2.28 where all the offset curves $\hat{\mathbf{c}}, \hat{\mathbf{c}}_{1/4}, \hat{\mathbf{c}}_{3/4}, \hat{\mathbf{c}}_{2/5}, \hat{\mathbf{c}}_{3/5}$ are almost coinciding thus the distance between the shoulder points of all these conics is negligible. As a measure of closeness, we propose to use the distance between the shoulder points of the three conics $\hat{c}_{1/4}, \hat{c}_{3/4}$ and $\hat{c}$.

If the shoulder points are too far apart, we *subdivide and standardize* $\mathbf{c}$ at some parameter value t and then repeat the previously described second method of constructing offsets for each of the subdivided conics. This defines a *recursive offset procedure* as seen from the *Figure*. It is convenient to subdivide the curve at $t = 1/2$, so that the simplifications becomes easier and new control polygons are given by $\mathbf{b}_0, \mathbf{q}_0, \mathbf{s}$ and $\mathbf{s}, \mathbf{q}_1, \mathbf{b}_2$. The weights for both the segments are $1, \sqrt{\frac{w+1}{2}}, 1$. Now to draw the offsets for each of the subdivide curves the offsets of $\mathbf{b}_0, \mathbf{q}_0, \mathbf{s}, \mathbf{q}_1, \mathbf{b}_2$ have to be found so that the control polygons for the new offset approximations are given by $\hat{\mathbf{b}}_0, \hat{\mathbf{b}}_0^1, \hat{\mathbf{s}}$ and $\hat{\mathbf{s}}, \hat{\mathbf{b}}_1^1, \hat{\mathbf{b}}_2$. Now their weights have to be determined by using the equation (2.27) and the procedure explained in the second method of constructing offsets. This can be seen from the *Figure* 2.11 where first the measure of closeness between the shoulder points of the conics $\hat{\mathbf{c}}_{1/4}, \hat{\mathbf{c}}_{3/4}, \hat{\mathbf{c}}_{2/5}, \hat{\mathbf{c}}_{3/5}$ is checked to be less than 0.05. If it is not so then we resort to the subdivision and standardization recursive offset procedure.

Offsets of curvature continuous piecewise rational quadratics: Till now we have been constructing the offsets of a single segment of a conic. Now let us see the offsets for a curvature continuous conic.

First the curvature continuous conic is drawn as explained previously in the corresponding section where the equation (2.22) is derived. Then for each segment its offset is drawn by using either of the two methods of drawing offsets. As seen in the *Figures* 2.29 and 2.30 there are three curves drawn with four segments for each of the curves drawn in such a way that the C^2 continuity is maintained. The middle curve is the original curve and the other two curves below and above it are its offsets. In the *Figure* 2.29 the points $\mathbf{b_0}, \mathbf{b_1}. \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$ are the given control points and the other control points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ are found such that they lie on the mid-point between the given control points. In the *Figure* 2.30 the points $\mathbf{b_0}, \mathbf{b_1}, \mathbf{b_3}, \mathbf{b_5}, \mathbf{b_7}$ are the given control points and the other control points $\mathbf{b_2}, \mathbf{b_4}, \mathbf{b_6}, \mathbf{b_8}$ are found such that they doesn't lie on the mid-point between the given control points and as seen the curve gets attracted towards the control polygon and finally coincides with the control polygon. As described previously, for each segment the offset can be drawn by using the first method where the offset of each point on the conic is drawn or by using the second method where for each segment the offset control polygons are found and an approximate weight for the first segment of offset is found by the procedure described previously for a single segment, then the weights of other segments of offsets are found by using equation (2.22). Then once all weights and

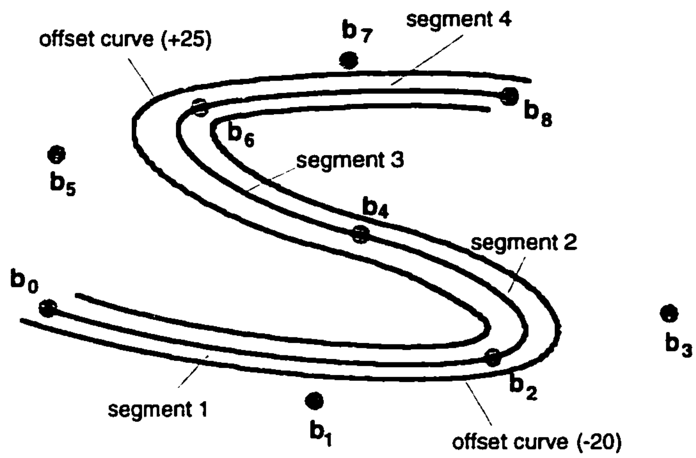


Figure 2.29: curvature continuous piecewise rational quadratics with offsets; junction points at mid-points; initial weight given for the first segment is 1.5

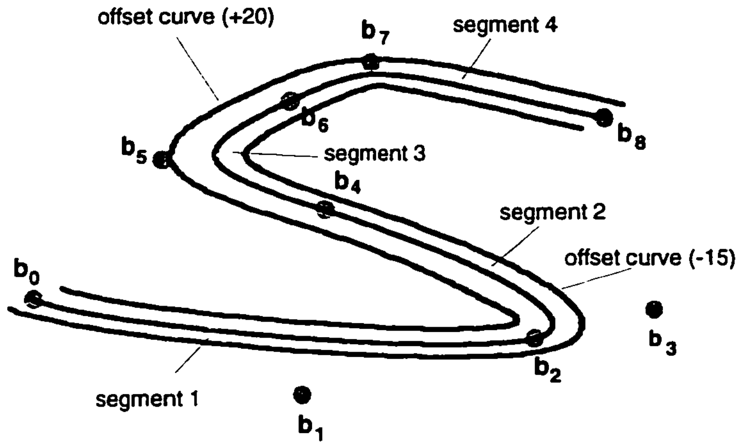


Figure 2.30: curvature continuous piecewise rational quadratics with offsets; junction points not at mid-points; initial weight given for the first segment is 0.9

control polygons for each segment is obtained the offset curvature continuous conic is drawn.

Chapter 3

Representation of a C^1 continuous rational cubic and its alternate C^1 continuous rational quadratic

3.1 C^1 rational cubic interpolant

A piecewise rational cubic parametric function $S \in C^1[t_1, t_{n+1}]$, with parameters $r_i, i = 1, ..., n$, is defined for $t \in [t_i, t_{i+1}], i = 1, ..., n$, by

$$S(t) = S_i(t) = \frac{F_i(1-\theta)^3 + r_i U_i \theta(1-\theta)^2 + r_i V_i \theta^2(1-\theta) + F_{i+1}\theta^3}{(1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3} \tag{3.1}$$

where the $F_i \in \mathbf{R}^N$ are the data values (given control points) at the knots $t_i, i = 1, ..., n+1$ with $t_1 < t_2 <, ..., < t_{n+1}, h_i = t_{i+1} - t_i, \theta = \frac{(t-t_i)}{h_i}$ and $r_i \geq 0$ is the shape

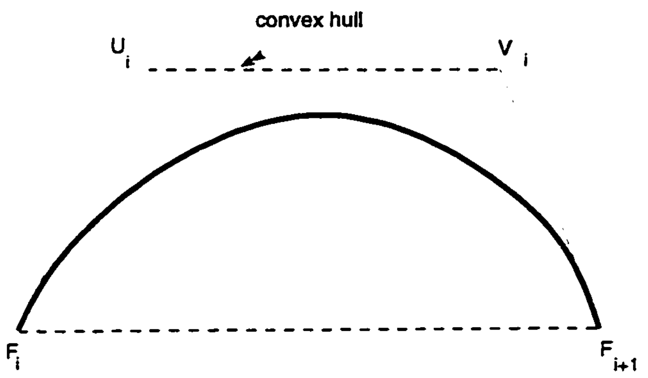


Figure 3.1: Rational Cubic.

control parameter for segment i. The U_i and V_i's are the *intermediate control points* (*see Figure 3.1*); with the properties $S(t_i) = F_i$ and $S^{'}(t_i) = D_i, i = 1, ..., n+1$ where $D_i \in \mathbf{R}^N$ are the first derivative values at the knots $t_i, i = 1, ..., n+1$ as done in [33]. *As seen from the equation U_i and V_i need to be found while other parameters are given. The curve segment varies between the parameter values t_i and t_{i+1} depending on the shape control parameter r_i (see Figure 3.2) and the derivatives at the control points* (*see Figures 3.3, 3.4, and 3.5*). The values to be taken for D_i's is discussed later in the chapter.

Since the C^1 continuity is being discussed, consider the *first derivative* of $S_i(t)$ of equation (3.1), which is given as:

$$S_i^{'}(t) = \frac{r_i(U_i - F_i)(1-\theta)^4 + (3(F_i + F_{i+1}) + r_i^2(V_i - U_i))\theta^2(1-\theta)^2}{h_i[\mathbf{D}]^2}$$

$$+\frac{2r_i(V_i - F_i)\theta(1-\theta)^3 + 2r_i(F_{i+1} - U_i)\theta^3(1-\theta) + r_i(F_{i+1} - V_i)\theta^4}{h_i[\mathbf{D}]^2} \qquad (3.2)$$

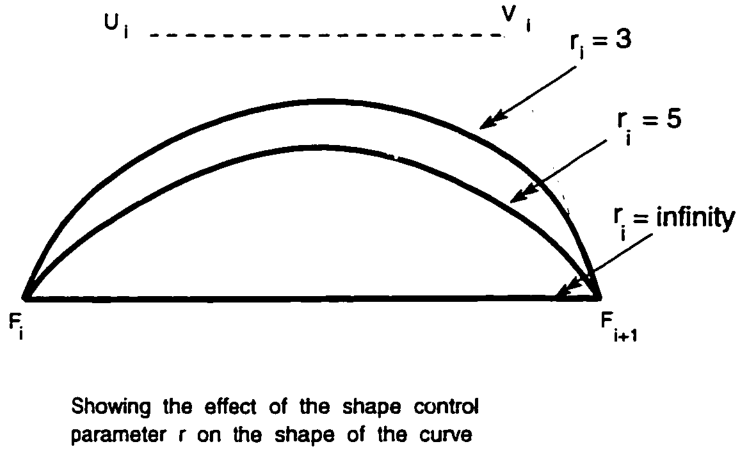


Figure 3.2: Rational Cubic : *showing the effect of shape control parameter on the shape of the curve.*

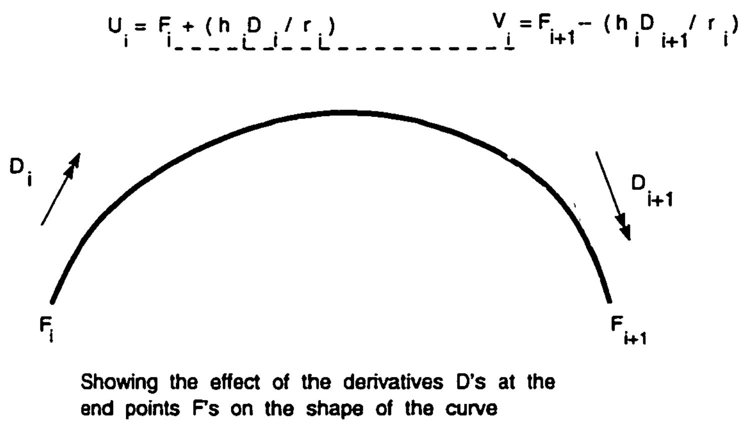
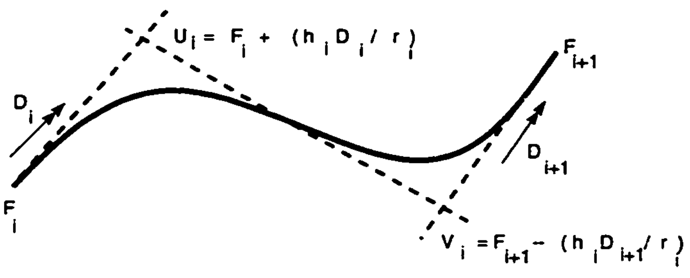
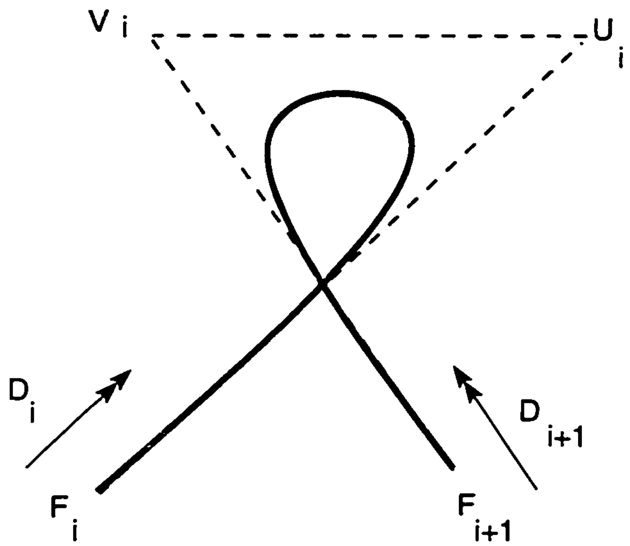


Figure 3.3: Rational Cubic : *showing the effect of the derivatives (at the end points) on the shape of the curve.*



Showing the effect of the derivatives D's at the end points F's on the shape of the curve

Figure 3.4: Rational Cubic : *showing the effect of the derivatives (at the end points) on the shape of the curve.*



Showing the effect of the derivatives D's at the end points F's on the shape of the curve

Figure 3.5: Rational Cubic : *showing the effect of the derivatives (at the end points) on the shape of the curve.*

where **D** is the denominator of equation (3.1).

If $t = t_i$, θ *becomes* 0 and the equation (3.2) reduces to

$$S_i'(t_i) = \frac{r_i(U_i - F_i)}{h_i} \tag{3.3}$$

Since $S_i'(t_i) = D_i$, the equation (3.3) can now be written as

$$\frac{r_i(U_i - F_i)}{h_i} = D_i$$

and thus the control point U_i can now be obtained as

$$U_i = F_i + \frac{h_i D_i}{r_i} \tag{3.4}$$

Similarly, if $t = t_{i+1}$, θ *becomes* 1 and the equation (3.2) reduces to

$$S_i'(t_{i+1}) = \frac{r_i(F_{i+1} - V_i)}{h_i} \tag{3.5}$$

Since $S_i'(t_{i+1}) = D_{i+1}$, the equation (3.5) can now be written as

$$\frac{r_i(F_{i+1} - V_i)}{h_i} = D_{i+1}$$

and thus the control point V_i can now be obtained as

$$V_i = F_{i+1} - \frac{h_i D_{i+1}}{r_i} \tag{3.6}$$

Using the equations (3.4) and (3.6) for U_i and V_i respectively, the function $S(t)$ in the *Hermite interpolation form* is given as :

$$S(t) = S_i(t) = \frac{F_i(1-\theta)^3 + (r_i F_i + h_i D_i)\theta(1-\theta)^2 + (r_i F_{i+1} - h_i D_{i+1})\theta^2(1-\theta) + F_{i+1}\theta^3}{(1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3} \tag{3.7}$$

The $r_i, i = 1, ..., n$, will be used so that the shape of the curve is controlled. The case $r_i = 3, i = 1, ..., n$, is that of *cubic Hermite interpolation* and the restriction, $r_i \geq 0$ ensures a *positive denominator* in equations (3.1) and (3.7).

For $r_i \neq 0$, equation (3.7) can be written in the form

$$S_i(t_i; r_i) = R_0(\theta; r_i)F_i + R_1(\theta; r_i)U_i + R_2(\theta; r_i)V_i + R_3(\theta; r_i)F_{i+1}$$

where U_i and V_i are as given in equation (3.4) and (3.6) respectively and $R_j(\theta; r_i), j = 0, 1, 2, 3$, are appropriately defined rational functions with

$$\sum_{j=0}^{3} R_j(\theta; r_i) = 1$$

Moreover, these functions are rational Bernstein-Bezier weight functions, which are *non-negative for* $r_i > 0$ as described in [33]. That is for $R^N, N > 1$ and for $r_i > 0$, we have :

- *Convex hull property.* The curve segment S_i lies in the convex hull of the control points F_i, U_i, V_i, F_{i+1} as seen in *Figure 3.1.*

- *Variation diminishing property.* The curve segment S_i crosses any (hyper) plane of dimension $N - 1$ no more often than it crosses the control polygon joining F_i, U_i, V_i, F_{i+1} *(see Figure 3.6).* It was proved in [34].

Assume the shape parameter $r_i > 0$, then for given fixed (or bounded) D_i and D_{i+1}, the following observations can be made immediately from the Bernstein-Bezier representation, and the control points U_i and V_i defined by equations (3.4) and (3.6)

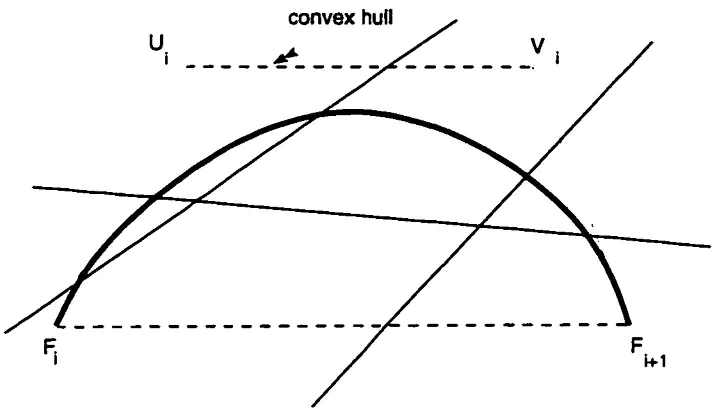


Figure 3.6: Rational Cubic : *showing the variation diminishing property.*

- If $r_i \to \infty$, then the rational cubic Hermite interpolant equation(3.7) converges to the *linear interpolant* $L_i(t)$ where

$$L_i(t) = (1-\theta)F_i + \theta F_{i+1} \tag{3.8}$$

  i.e., the increase in r_i **pulls** the curve towards F_i and F_{i+1} in the interval $[t_i, t_{i+1}]$ and the interpolant is linear as seen in *Figure 3.2.*

Now consider the effect of the shape parameter on the rational spline interpolant :

- *Global tension property.* Let $L \in C^0[t_1, t_{n+1}]$ denote the piecewise linear interpolant defined for $t \in [t_i, t_{i+1}]$ by

$$L(t) = L_i(t)$$

  see Eq.(8). Suppose that D_i's are bounded, then the rational spline interpolant **converges uniformly** to L as $r \to \infty$, *i.e.*, on $[t_1, t_{n+1}]$, when $r \to \infty$, $||S - L|| = 0$. See *Figure 3.7(a).*

- *Interval tension property.* Let D_i's are bounded and consider an interval $[t_k, t_{k+1}]$ for a fixed $k \in 1, ..., n$. Then on $[t_k, t_{k+1}]$ with fixed $r_i, i \neq k$, when $r_k \rightarrow \infty, ||S_k - L_k|| = 0$ See *Figure 3.8(a).*

The rational spline interpolant on $[t_k, t_{k+1}]$ converges uniformly to the line segment l_k (see *Interval tension property*) as $r \rightarrow \infty$. Moreover, for the range $0 < r_i < 3$ the rational spline produces a more flexible *i.e., looser* curve than the cubic spline curve, both locally and globally as shown in *Figure 3.7(b).* In addition to the assumptions in the previous statement, if it is also assumed that $r_{k-1} \rightarrow \infty$, then the curve at the point P_k will appear to have a *corner* as seen from *Figure 3.8(b).* All these properties were proved in [33] and [34].

For the rational cubic curve $S(t)$ to be C^1 continuous, the piecewise segment $S_i(t)$ passing through the points F_i and F_{i+1} with first derivatives D_i and D_{i+1} respectively at these points and the next segment $S_{i+1}(t)$ passing through the points $\hat{F}_{i+1}$ and F_{i+2} with first derivatives $\hat{D}_{i+1}$ and D_{i+2} respectively at these points, be such that the point F_{i+1} (the end point of segment $S_i(t)$) and the point $\hat{F}_{i+1}$ (the starting point of segment $S_{i+1}(t)$) should coincide and the first derivatives at this point D_{i+1} and $\hat{D}_{i+1}$ respectively should be same that is, $F_{i+1} = \hat{F}_{i+1}$ and $D_{i+1} = \hat{D}_{i+1}$.

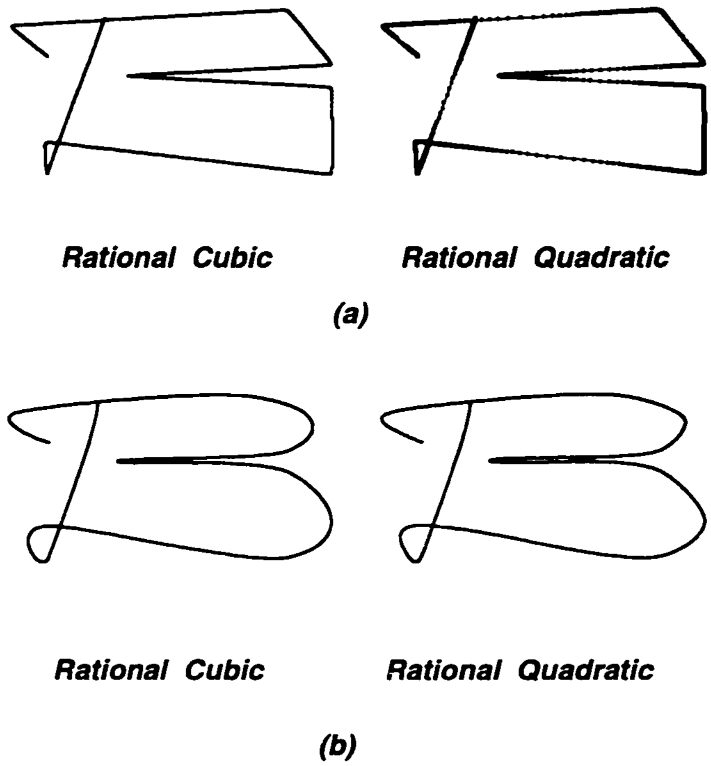


Figure 3.7: C^1 continuous rational cubic and rational quadratic : (a) *Global tension property*; (b) *Looser curve case* ; uniform h_i's (=1) are taken for both figures.

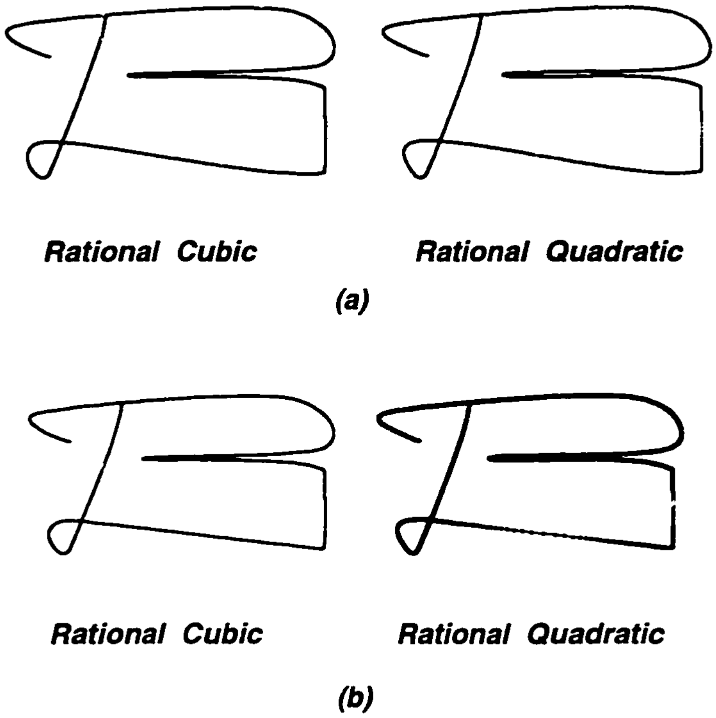


Figure 3.8: C^1 continuous rational cubic and rational quadratic : (a) *Interval tension property*; (b) *The corner effect*; uniform h_i's (=1) are taken for both figures.

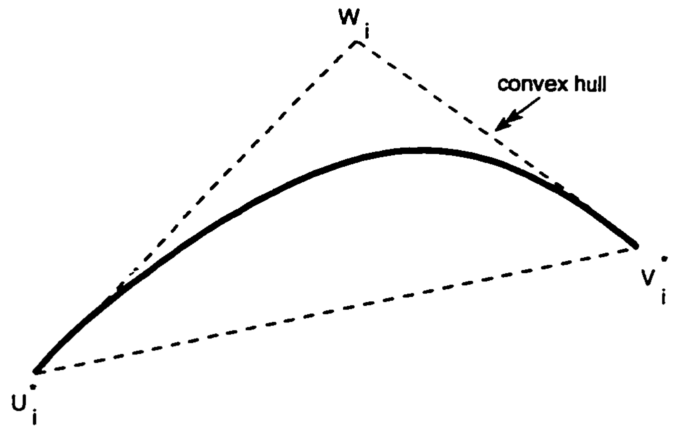


Figure 3.9: Conic.

3.2 C^1 rational quadratic interpolant

Consider the rational quadratic, defined as (*see Figure 3.9*) :

$$P_i(t) = \frac{U_i^*(1-\hat{\theta})^2 + \gamma_i W_i \hat{\theta}(1-\hat{\theta}) + V_i^* \hat{\theta}^2}{(1-\hat{\theta}^2) + \gamma\hat{\theta}(1-\hat{\theta}) + \hat{\theta}^2} \tag{3.9}$$

where for a particular segment i, the U_i^*, W_i and V_i^* are the three control points such that the conic passes through two points U_i^* and V_i^* and the point W_i affects the shape of the curve; γ_i is the *shape control parameter* for the segment i.

To get an alternate rational quadratic representation, each segment of the piecewise rational cubic curve has to be considered separately and each segment of the rational cubic can be represented by two segments of the rational quadratic.

Consider the rational cubic spline segment given by equation (3.1). To represent this rational cubic by an alternate conic representation three more points U_i^*, V_i^* and W_i in terms of the given points F_i, F_{i+1} and derivatives D_i and D_{i+1} have to

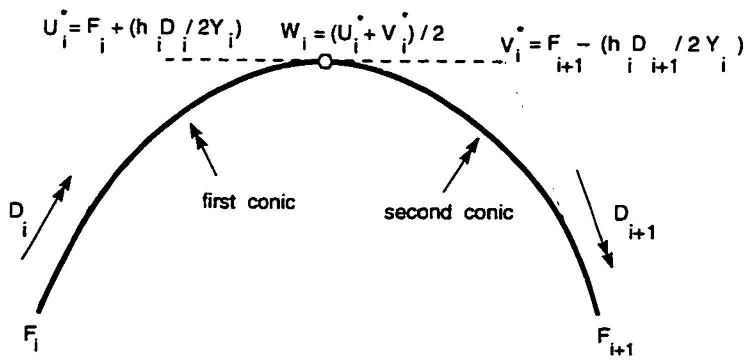


Figure 3.10: A Rational Cubic approximated by two Conics.

be found. These three points U_i^*, V_i^*, W_i along with F_i and F_{i+1} form the control points of the two conic segments representing the given rational cubic segment. *The point W_i is common to both the conics such that it is the end point of one conic and starting point of another conic*, as shown in *Figure 3.10.*

Let us consider the first conic segment which lies in the convex hull of F_i, U_i^* and W_i

$$P_i(t) = \frac{F_i(1-\theta)^2 + \gamma_i U_i^* \theta(1-\theta) + W_i\theta^2}{1 + (\gamma_i - 2)\theta(1-\theta)}$$

or equivalently as

$$P_i(t) = \frac{F_i(1-\theta)^2 + \gamma_i U_i^* \theta(1-\theta) + W_i\theta^2}{(1-\theta)^2 + \gamma_i\theta(1-\theta) + \theta^2} \tag{3.10}$$

where $\theta \equiv \theta(t) = \frac{2\times(t-t_i)}{h_i}$, $t\epsilon[t_i, t_i^*]$, and $t_i^* = \frac{(t_i+t_{i+1})}{2}$, such that t_i is the value of the parameter t at the starting point F_i of the first of the two conic segments that represent the given rational cubic segment and t_{i+1} is the value of the parameter t at the end point F_{i+1} of the second of the two conic segments that represent the given rational cubic segment; t_i^* *is the value of the parameter* t *at the* **mid-point** *of* t_i *and* t_{i+1} *which is the coinciding point* W_i *of the two conic segments.*

After simplifications the first derivative of $P_i(t)$ is calculated as :

$$P_i'(t) = 2 \times [\frac{r_i(U_i^* - F_i)(1-\theta)^3 + (2W_i + r_iU_i^* - r_iF_i - 2F_i)\theta(1-\theta)^2}{h_i[\mathbf{D1}]^2}$$

$$+\frac{(r_iW_i + 2W_i - r_iU_i^* - 2F_i)\theta^2(1-\theta) + r_i(W_i - U_i^*)\theta^3}{h_i[\mathbf{D1}]^2}] \qquad (3.11)$$

where $[\mathbf{D1}]$ is the Denominator of equation (3.10).

It can be noted that *when* $t = t_i$, *that is, at the starting point of the first conic,* $\theta(t_i)$ *becomes* 0 and hence $P_i'(t)$ is

$$P_i'(t_i) = \frac{2\gamma_i(U_i^* - F_i)}{h_i} \qquad (3.12)$$

and *when* $t = t_i^*$, *that is, at the end point of the first conic,* $\theta(t_i^*)$ *becomes* 1 and hence $P_i'(t)$ is

$$P_i'(t_i^*) = \frac{2\gamma_i(W_i - U_i^*)}{h_i} \qquad (3.13)$$

Now, consider the second conic which lies within the control points W_i, V_i^* and F_{i+1}

$$P_i^*(t) = \frac{W_i(1-\theta^*)^2 + \gamma_iV_i^*\theta^*(1-\theta^*) + F_{i+1}\theta^{*2}}{1 + (\gamma_i - 2)\theta^*(1-\theta^*)}$$

or equivalently as

$$P_i^*(t) = \frac{W_i(1-\theta^*)^2 + \gamma_iV_i^*\theta^*(1-\theta) + F_{i+1}\theta^{*2}}{(1-\theta^*)^2 + \gamma_i\theta^*(1-\theta^*) + \theta^{*2}} \qquad (3.14)$$

where $\theta^* \equiv \theta^*(t) = \frac{2\times(t-t_i^*)}{h_i}, t\epsilon[t_i^*, t_{i+1}]$.

The first derivative of $P_i^{*}(t)$, after simplifications, is obtained as :

$$P_i^{*'}(t) = 2 \times [\frac{r_i(V_i^{*} - W_i)(1-\theta^{*})^3 + (2F_{i+1} + r_iV_i^{*} - r_iW_i - 2W_i)\theta^{*}(1-\theta^{*})^2}{h_i[\mathbf{D2}]^2}$$

$$+ \frac{(r_iF_{i+1} + 2F_{i+1} - r_iV_i^{*} - 2W_i)\theta^{*2}(1-\theta^{*}) + r_i(F_{i+1} - V_i^{*})\theta^{*3}}{h_i[\mathbf{D2}]^2}] \quad (3.15)$$

where $[\mathbf{D2}]$ is the Denominator of equation (3.14).

One can observe that *when $t = t_i^{*}$, that is, at the starting point of the second conic, $\theta(t_i^{*})$ becomes* 0 and hence $P_i'(t)$ is

$$P_i^{*'}(t_i^{*}) = \frac{2\gamma_i(V_i^{*} - W_i)}{h_i} \quad (3.16)$$

and *when $t = t_{i+1}$, that is, at the end point of the second conic, $\theta(t_{i+1})$ becomes* 1 and hence $P_i'(t)$ is

$$P_i^{*'}(t_{i+1}) = \frac{2\gamma_i(F_{i+1} - V_i^{*})}{h_i} \quad (3.17)$$

In order that the conic representation is C^1, the following can be noted :

- The first derivative of the first conic segment $P_i'(t)$ at the starting point where the parameter $t = t_i$ should be equal to the first derivative of the rational cubic segment at its starting point where $t = t_i$, that is,

  $$P_i'(t_i) = D_i$$

  together with equation (3.12) implies

  $$U_i^{*} = F_i + \frac{h_iD_i}{2\gamma_i} \quad (3.18)$$

- The first derivative of the second conic segment $P_i^{*\prime}(t)$ at the end point where the parameter $t = t_{i+1}$ should be equal to the first derivative of the rational cubic segment at its end point where $t = t_{i+1}$, that is,

$$P_i^{*\prime}(t_{i+1}) = D_{i+1}$$

together with equation (3.17) implies

$$V_i^* = F_{i+1} - \frac{h_i D_{i+1}}{2\gamma_i} \tag{3.19}$$

- The first derivative of the first conic segment $P_i'(t)$ at its end point where the parameter $t = t_i^*$ should be equal to the first derivative of the second conic segment at its starting point which coincides with the end point of the first conic segment and where $t = t_i^*$, that is,

$$P_i'(t_i^*) = P_i^{*\prime}(t_i^*)$$

together with equation (3.13) and (3.16) implies

$$W_i = \frac{U_i^* + V_i^*}{2}$$

From equations (3.18) and (3.19)

$$W_i = \frac{2\gamma_i F_i + h_i D_i + 2\gamma_i F_{i+1} - h_i D_{i+1}}{4\gamma_i}$$

$$W_i = \frac{F_i + F_{i+1}}{2} + \frac{h_i}{4\gamma_i}(D_i - D_{i+1}) \tag{3.20}$$

Now that the unknown three points U_i^*, V_i^* and W_i are found in terms of F_i, F_{i+1}, D_i, D_{i+1}, γ_i, and t_i's they can be substituted in equations (3.10) and (3.14) to get the required conic representation of rational cubic given by equation (3.1). **Thus each segment of a piecewise rational cubic is represented by two segments of conics, and by giving different points F_i, F_{i+1} and parameter values t_i and r_i's for each segment we get a piecewise rational cubic which can alternatively be represented by a piecewise conic** (*see Figure 3.10*).

The properties that are considered for rational cubic are also satisfied by the conic

- *Convex hull property.* The curve segments $P_i(t)$ and $P_i^*(t)$ lies in the convex hull of the control points F_i, U_i^*, W_i and W_i, V_i^*, F_{i+1} respectively as seen in *Figure 3.9.*

- *Variation diminishing property.* The curve segments P_i and $P_i^*(t)$ crosses any (hyper) plane of dimension $N-1$ no more often than it crosses the control polygon joining F_i, U_i^*, W_i and W_i, V_i^*, F_{i+1} respectively.

Here also the *increase in γ_i pulls the first conic towards F_i, W_i in the interval $[t_i, t_i^*]$ and pulls the second conic towards W_i, F_{i+1} in the interval $[t_i^*, t_{i+1}]$*. The *global tension* and *interval tension* properties are also satisfied by the conics. Moreover, for the range $0 < \gamma_i < 2$ the rational quadratic produces a more flexible *i.e., looser*, curve than the quadratic polynomial, both locally and globally.

3.3 Choice of D_i's

An approximate choice of first derivatives may be given by the following formula :

$$D_i = \frac{h_{i-1}\Delta_i + h_i\Delta_{i-1}}{h_i + h_{i-1}} \tag{3.21}$$

where

$$\Delta_i = \frac{F_{i+1} - F_i}{h_i}$$

and

$$h_i = t_{i+1} - t_i$$

for $i = 2, ..., n$. The D_1 and D_{n+1} are considered as different for the open and closed curve cases.

For the *open curve* case they are given as :

$$D_1 = \frac{\Delta_1 + (\Delta_1 - \Delta_2)h_1}{h_1 + h_2} \tag{3.22}$$

$$D_{n+1} = \frac{\Delta_{n-1} + (\Delta_n - \Delta_{n-1})h_{n-1}}{h_n + h_{n-1}} \tag{3.23}$$

or they can be given as desired by the user.

For the *closed curve* case they are obtained from equation (3.21) as :

$$D_1 = \frac{h_0\Delta_1 + h_1\Delta_0}{h_0 + h_1}$$

where

$$\Delta_0 = \frac{F_1 - F_0}{h_0}, h_0 = t_1 - t_0.$$

Since, in closed curve case $F_1 = F_{n+1}$, it can be said that

$$F_0 = F_n, t_0 = t_n, t_1 = t_{n+1};$$

This results in

$$\Delta_0 = \frac{F_1 - F_n}{t_1 - t_n}$$

$$\Delta_0 = \frac{F_{n+1} - F_n}{t_{n+1} - t_n}$$

which is equal to Δ_n from equation (3.21) and $h_0 = t_{n+1} - t_n$ which is equal to h_n. So, D_1 now becomes

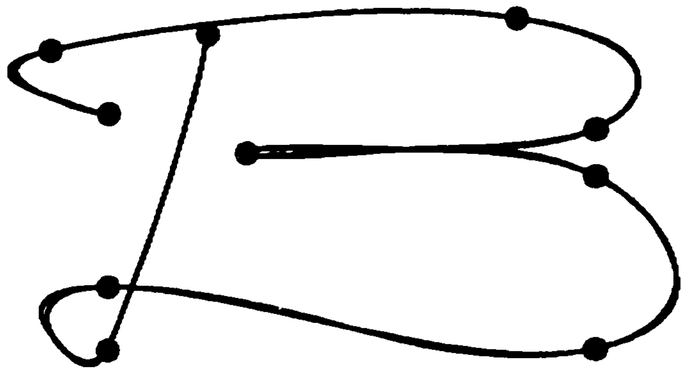
$$D_1 = \frac{h_n \Delta_1 + h_1 \Delta_n}{h_1 + h_n} \tag{3.24}$$

and since in closed curve $D_1 = D_{n+1}$, the equation for D_{n+1} is same as equation (3.24).

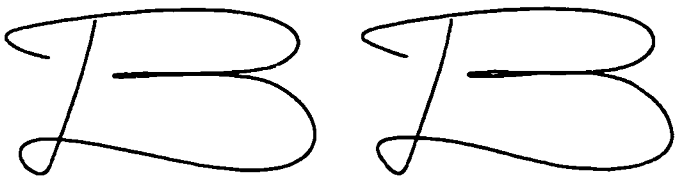
In addition to the *arithmetic mean* choice of D_i's as in equation (3.21), some other approximate choices like *geometric mean or harmonic mean* choices, are also available. However, if the D_i's are calculated from the description of the C^2 cubic splines, there may be an achievement in the betterment of smoothness. Such D_i's are calculated from the following system of equations :

$$h_{i+1} D_i + (h_i(r_{i+1} - 1) + h_{i+1}(r_i - 1)) D_{i+1} + h_i D_{i+2} = h_{i+1} r_i \Delta_i + h_i r_{i+1} \Delta_{i+1}$$

which is a diagonally dominant tridiagonal system of equations in unknown D_i's. An appropriate choice of end conditions D_1 and D_{n+1} will provide a unique solution to an appropriate spline case.



Rational Cubic and Rational Quadratic



Rational Cubic ***Rational Quadratic***

(of the above representation)

Figure 3.11: C^1 continuous rational cubic and rational quadratic : C^2 *continuity derivatives are used*; default values for shape parameters and uniform h_i's (=1) are taken.

3.4 Closed curves

Till now the C^1 continuous open curve were such that the tangent vectors D_i's were independent of one another and were completely dependent on the control points and the range of the parameter t. Now, consider the C^1 continuous closed curves where the first and last control points should coincide and also the initial and final derivatives should be equal. That is, if there are $n+1$ control points of an n segment curve then for the curve to be closed with C^1 continuity the control points F_1 and F_{n+1} should be same and also the initial derivative D_1 and final derivative D_{n+1} should be equal.

$$F_1 = F_{n+1}, D_1 = D_{n+1}$$

If the above conditions are satisfied a *cyclic closed curve* is obtained to produce a closed curve or a portion of a curve that repeats at intervals.

The *anticyclic closed curve* is similar to the cyclic closed curve except that

$$D_1 = -D_{n+1}$$

The anticyclic curve are useful for producing parallel end spans with end tangent vectors that are *equal in magnitude but opposite in direction.* A practical example is a laminated wooden tennis racket.

By using the above conditions for cyclic and anticyclic curves a given closed rational cubic given by equation (3.1) can alternatively be represented by a closed

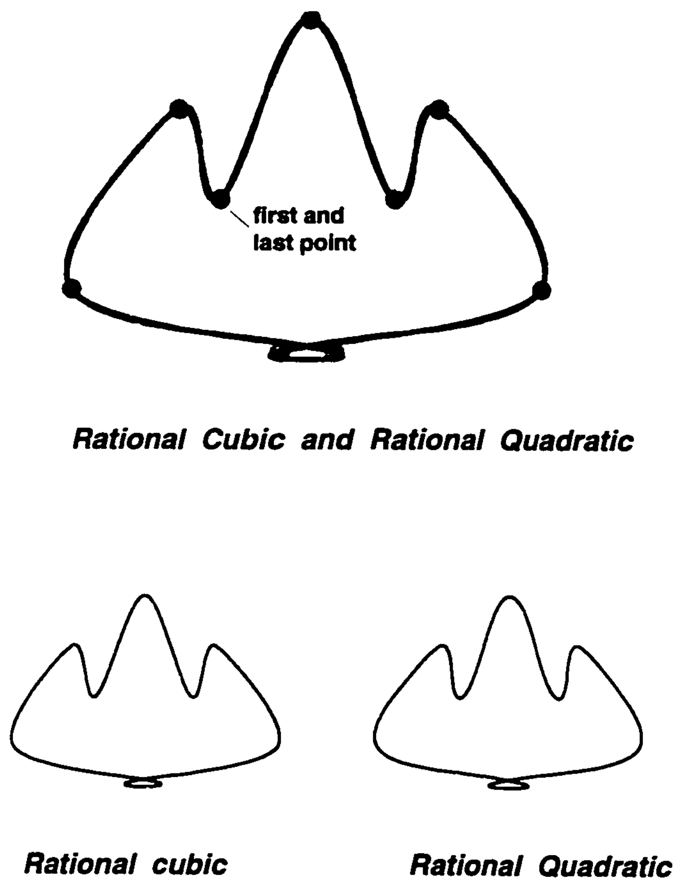
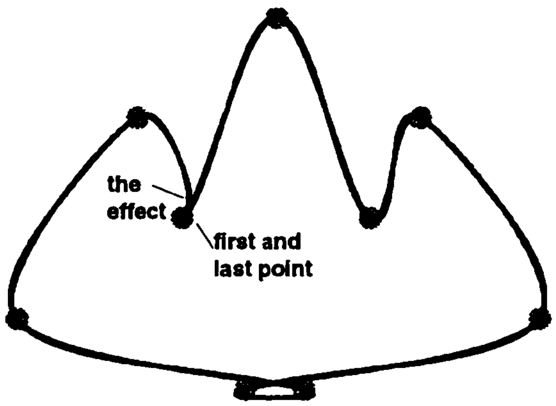
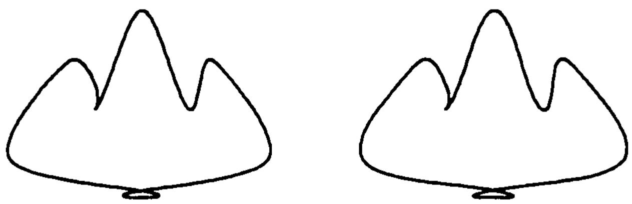


Figure 3.12: C^1 continuous rational cubic and rational quadratic : *cyclic closed curve case*; default values for shape parameters and uniform h_i's (=1) are taken.



Rational Cubic and Rational Quadratic



Rational Cubic ***Rational Quadratic***

(of the above representation)

Figure 3.13: C^1 continuous rational cubic and rational quadratic : *anticyclic closed curve case*; default values for shape parameters and uniform h_i's (=1) are taken.

conic given by equations (3.10) and (3.14). However, since here a rational cubic segment is represented by two conic segments, if the D_1 of the first rational cubic segment is equal to the D_{n+1} of the last rational cubic segment then the D_1 of the first of the two of the conic segments representing the first rational cubic segment is equal to the D_{n+1} of the second of the two of the conic segments representing the last rational cubic segment.

After getting an alternate conic representation, for a given rational cubic representation, now consider some special cases of their representations.

3.5 Default case

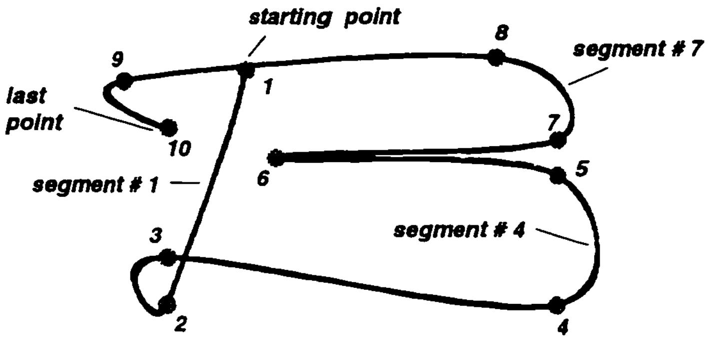
If the value of the shape parameter r_i of rational cubic is taken as 3, then the denominator of equation (3.1) becomes 1 and the equation (3.1) becomes

$$S_i(t) = F_i(1-\theta)^3 + 3U_i\theta(1-\theta)^2 + 3V_i\theta^2(1-\theta) + F_{i+1}\theta^3 \tag{3.25}$$

which is no more rational cubic but simply a cubic.

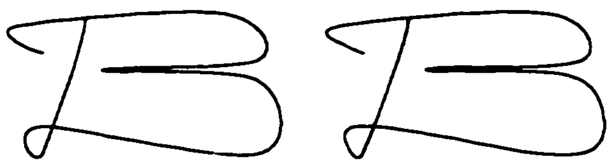
Similarly, if the value of the shape parameter γ_i of conic is taken as 2, then the denominator of equations (3.10) and (3.14) becomes 1 and the equations (3.10) and (3.14) become

$$P_i(t) = F_i(1-\theta)^2 + 2U_i^*\theta(1-\theta) + W_i\theta^2 \tag{3.26}$$



Rational Cubic and Rational Quadratic

(to show how Rational Quadratic exactly matches the Rational Cubic)



Rational Cubic ***Rational Quadratic***

(of the above representation)

Figure 3.14: C^1 continuous rational cubic and rational quadratic : *Default case* with uniform h_i's $(=1)$ are taken.

and

$$P_i^*(t) = W_i(1-\theta)^2 + 2V_i^*\theta(1-\theta) + F_{i+1}\theta^2 \quad (3.27)$$

which are no more rational quadratics but simply quadratic.

The cases of taking r_i as 3 and γ_i as 2 will be considered as the default case.

3.6 Control points matching case

Now, consider the case when the

$$\gamma_i = \frac{3}{2}$$

Considering equations (3.18) and (3.19) to get the control points of the conic

$$U_i^* = F_i + \frac{h_i D_i}{2\gamma_i}$$

and

$$V_i^* = F_{i+1} - \frac{h_i D_{i+1}}{2\gamma_i}$$

substituting $\gamma_i = \frac{3}{2}$ in the above two equations, results in

$$U_i^* = F_i + \frac{h_i D_i}{3}$$

and

$$V_i^* = F_{i+1} - \frac{h_i D_{i+1}}{3}$$

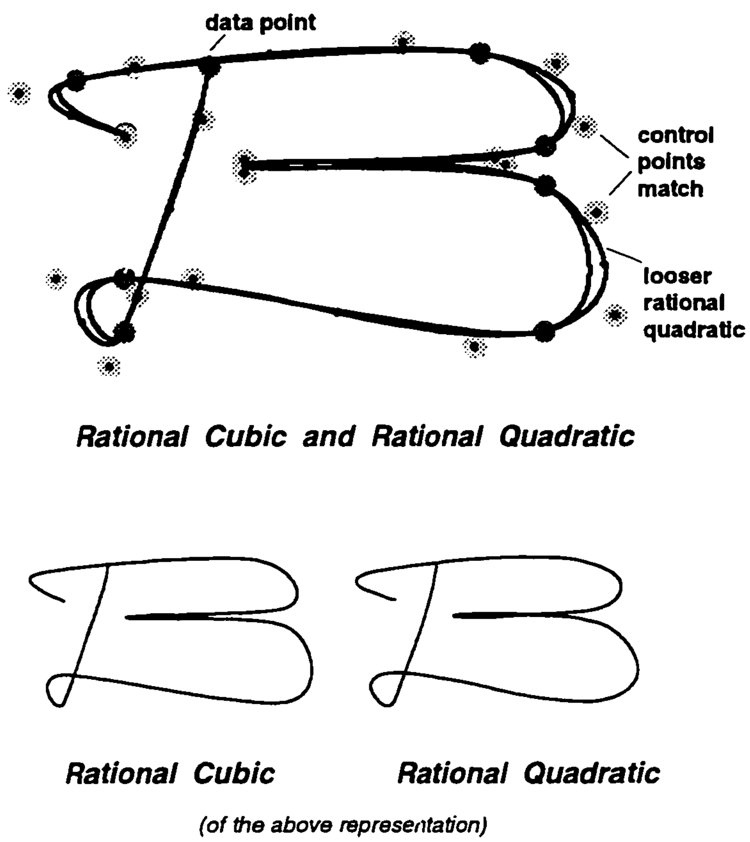


Figure 3.15: C^1 continuous rational cubic and rational quadratic : *control points matching case* where $r_i = 3.0$, $\gamma_i = 1.5$ *i.e.*, ($\gamma_i = \frac{r_i}{2}$); uniform h_i's (=1) are taken.

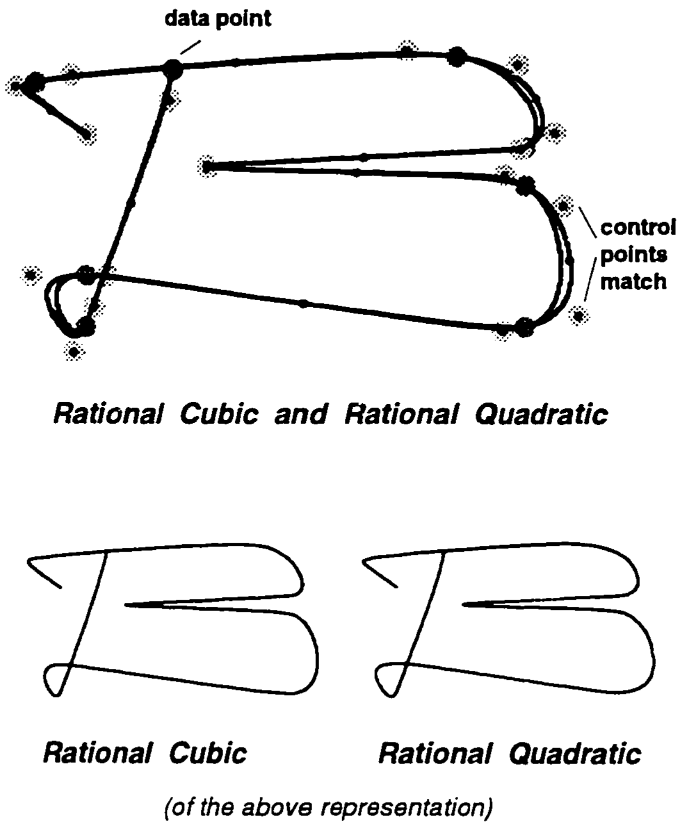


Figure 3.16: C^1 continuous rational cubic and rational quadratic : *control points matching case* where $r_i > 3.0$, $\gamma_i > 2.0$ *i.e.*, ($\gamma_i = \frac{r_i}{2}$); uniform h_i's (=1) are taken.

which are nothing but the equations (3.4) and (3.6) which are the control points of the rational cubic when default case is taken *i.e.*, when $r_i = 3$. Thus, if $\gamma_i = \frac{3}{2}$ is taken a condition is obtained where the two control points U_i^* and V_i^* of conics bf coincide with the control points U_i and V_i of the rational cubic.

Just as in the previous case if

$$\gamma_i = \frac{r_i}{2}$$

the overall best approximation of the rational cubic by conic is obtained, than it would have been had some other γ_i's been taken. When $\gamma_i = \frac{r_i}{2}$, the equations (3.18) and (3.19) becomes

$$U_i^* = F_i + \frac{h_i D_i}{r_i}$$

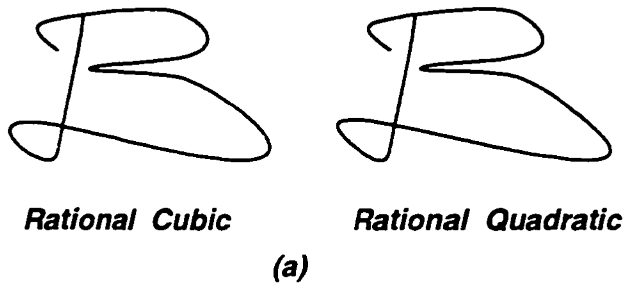
and

$$V_i^* = F_{i+1} - \frac{h_i D_{i+1}}{r_i}$$

which are simply the equations (3.4) and (3.6) of the control points of the rational cubic and thus the control points of both rational cubic and conics match.

3.7 The selection of t_i's

Recall that a piecewise rational cubic and conic curve is determined by the position vectors *i.e.*, the control points; the tangent vectors *i.e.*, the derivatives at these



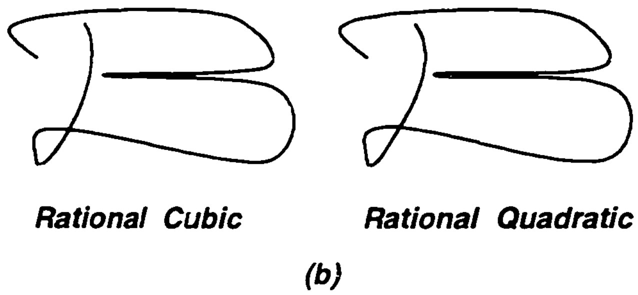


Figure 3.17: C^1 continuous rational cubic and rational quadratic : *non-uniform* h_i *'s case* where in (a) the t_i's are taken at different ranges; (b) the t_i's are taken as *chord length between the given points*; default values for shape parameters are taken for both figures.

control points; and the parameter values *i.e.*, the t_i's at the end of each segment. Just as the control points, the derivatives at these control points have effect on the curve's smoothness so does the values of the t_i's.

One approach used to determine the t_i's is to normalize the parameter values to unit distance by setting $t_1 = 0, t_2 = 1, t_3 = 2, ...t_n = n - 1$. A second approach could be to set the parameter values equal to the chord lengths between successive data points *i.e.*, t_i is equal to the distance between the F_i and F_{i+1} control points. Acceptably smooth curves for most practical purposes are obtained using this technique. As can be seen from the previous equations of rational cubic and conic, they are dependent on h_i apart from D_i's, r_i's and F_i's. The D_i's, as given by equation (3.21) are also dependent on h_i. Since these h_i's are obtained from t_i's, each choice of t_i's produces different coefficient values, and, hence different curves through the given data points.

3.8 Demonstration

In this section the various properties and cases discussed so far will be demonstrated for different type of curves (both open and closed). First, the effect of the shape parameter on the rational cubic and conic is considered. Let's start with open curve case. In *Figure 3.14*, the default values for the shape parameters are taken *i.e.*, $r_i = 3$ and $\gamma_i = 2; i = 1, ..., 9$ by taking the t_i's at unit distance *i.e.*, $t_{i+1} - t_i = 1; i = 1, ..., 10$.

so that all h_i's are 1. As seen from the figure the rational cubic representation is exactly matched with the conic representation. In *Figure 3.7(a)*, the effect of *global tension property* is shown by taking the r_i's as 50 and γ_i's as 25 *i.e.*, ($\gamma_i = \frac{r_i}{2}$). In *Figure 3.7(b)*, the $r_4 = r_7 = 1.5(< 3)$ and $\gamma_4 = \gamma_7 = 1.0(< 2)$ are taken and all other r_i's and γ_i's are kept at default values to show how looser curves are formed when the shape parameters are less than their default values. In *Figure 3.8(a)*, the effect of *interval tension property* is shown by taking r_4 as 50 and γ_4 as 25 and keeping all other r_i's at 3 and γ_i's at 2 and as seen from the figure the segment 4 is more tighter. In *Figure 3.8(b)*, the $r_3 = 50, r_4 = 50$ and similarly $\gamma_3 = 25, \gamma_4 = 25$ are taken and all other r_i's and γ_i's are kept at their default values, to show how a *corner* is formed at point P_4. In *Figure 3.15*, the r_i's are all kept at 3 and the γ_i's are all kept at 1.5 *i.e.*, $\gamma_i = \frac{r_i}{2}$ to show how the control points of both rational cubic and conic coincide. In *Figure 3.16*, $r_i > 3$ and $\gamma_i > 2$ and $\gamma_i = \frac{r_i}{2}$ for all i, and as seen the control points of both rational cubic and conic coincide in this case also.

Next, consider the effect of non-uniform h_i's *i.e.*, $t_{i+1} - t_i \neq t_{i+2} - t_{i+1}$ on the rational cubic and conic by keeping the values of shape parameters r_i and γ_i at their default as seen in *Figure 3.17(a)*. In *Figure 3.17(b)*, the h_i's are taken as chord length between F_i and F_{i+1} *i.e.*, $t_i = t_{i-1} + F_iF_{i+1}, i = 2, ..., n+1$ where F_iF_{i+1} is the distance between F_i and F_{i+1}.

As seen in all the figures demonstrated so far, the rational cubic is almost exactly represented by a rational quadratic or conic with very negligible deviation.

Now, let's consider the closed curve case. In *Figure 3.12*, a closed curve with *cyclic end condition*, default values for r_i and γ_i and uniform values for all $h_i (= 1)$ is taken. In *Figure 3.13*, a closed curve but with *anticyclic end condition*, default values for r_i and γ_i and uniform values for all $h_i (= 1)$ is taken and the effect is clearly seen. Also, all the cases considered so far for open curves - the control points matching case, nonuniform h_i's case, chord length values for t_i's, the global, interval tension properties; will also apply for the closed curves as well.

In *Figure 3.18* the default values for the shape parameters are taken *i.e.*, $r_i = 3$ and $\gamma_i = 2; i = 1, ..., 24$ by taking the t_i's at unit distance *i.e.*, $t_{i+1} - t_i = 1; i = 1, ..., 25$, so that all h_i's are 1. It demonstrates the closed curve case by showing an alphabet 'G'. In *Figure 3.19*, the $\gamma_i = \frac{r_i}{2}$ is taken such that $r_i \geq 3.0$ and $\gamma_i \geq 2.0$, h_i's are uniform (=1) and in this case the control points of both rational cubic and conic match. In *Figure 3.20*, r_i and γ_i are taken as chord length and h_i's are uniform (=1). In *Figure 3.21*, an alphabet 'S' is shown in which the $r_i \leq$ and ≥ 3.0 and the $\gamma_i \leq$ and ≥ 2.0 and h_i's are uniform. In *Figure 3.17*, the case where the derivatives taken from the C^2 continuous rational cubic (keeping r_i at 3) and used in the C^1 continuity, is considered.

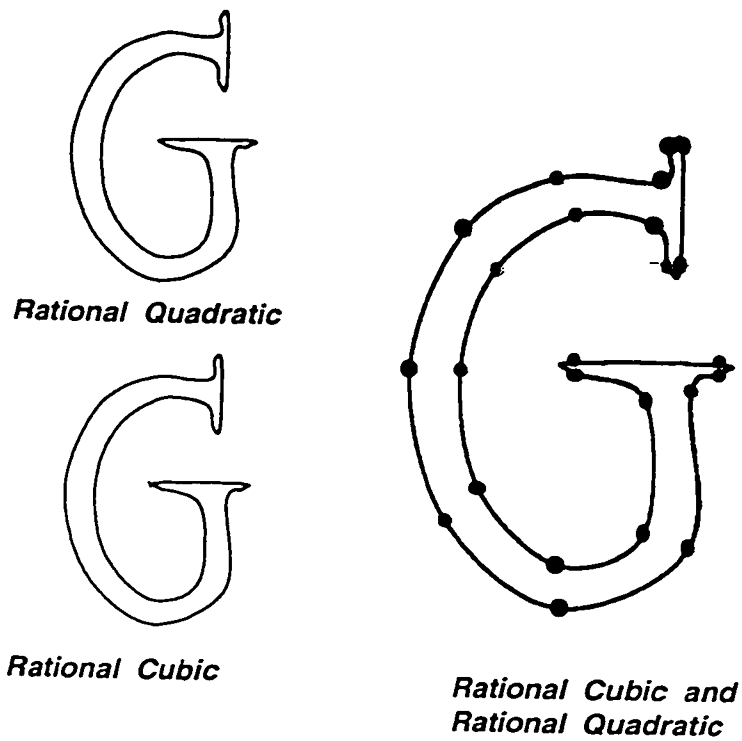


Figure 3.18: C^1 continuous rational cubic and rational quadratic : *an alphabet "G" is drawn*; default values for shape parameters and uniform h_i's $(=1)$ are taken.

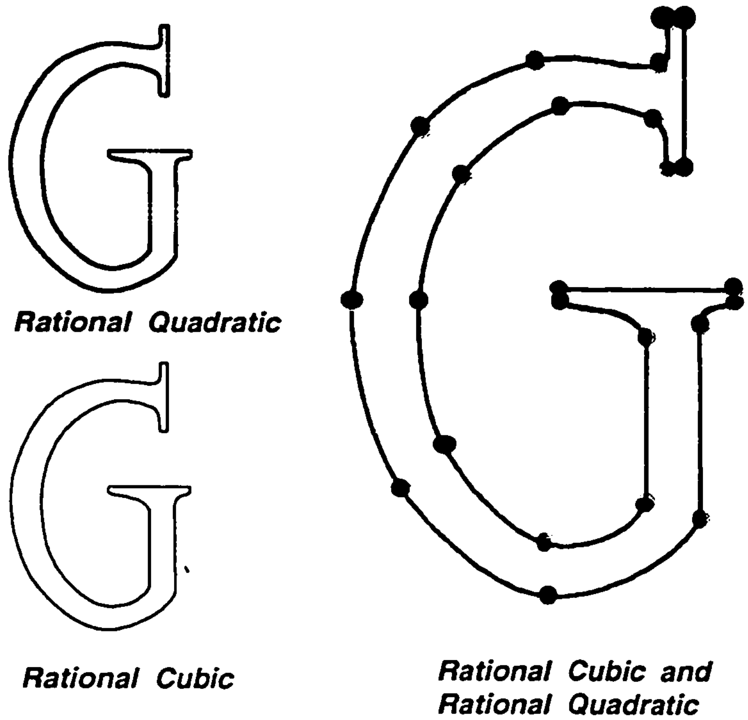


Figure 3.19: C^1 continuous rational cubic and rational quadratic : *an alphabet "G" is drawn*; where $r_i \geq 3.0, \gamma_i \geq 2.0$ *i.e.*, ($\gamma_i = \frac{r_i}{2}$); uniform h_i's (=1) are taken.

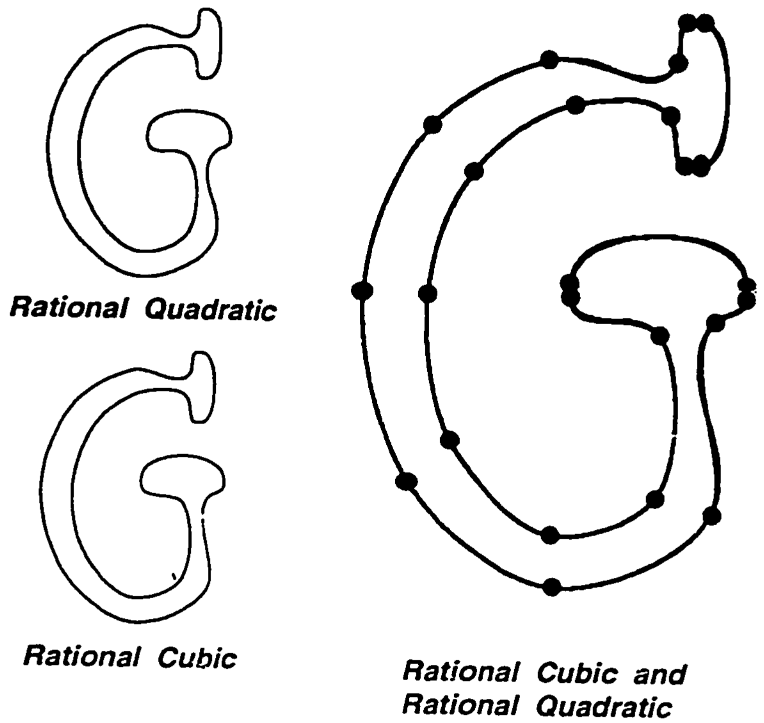


Figure 3.20: C^1 continuous rational cubic and rational quadratic : *an alphabet "G" is drawn*; default values for shape parameters and chord length for t_i's are taken.

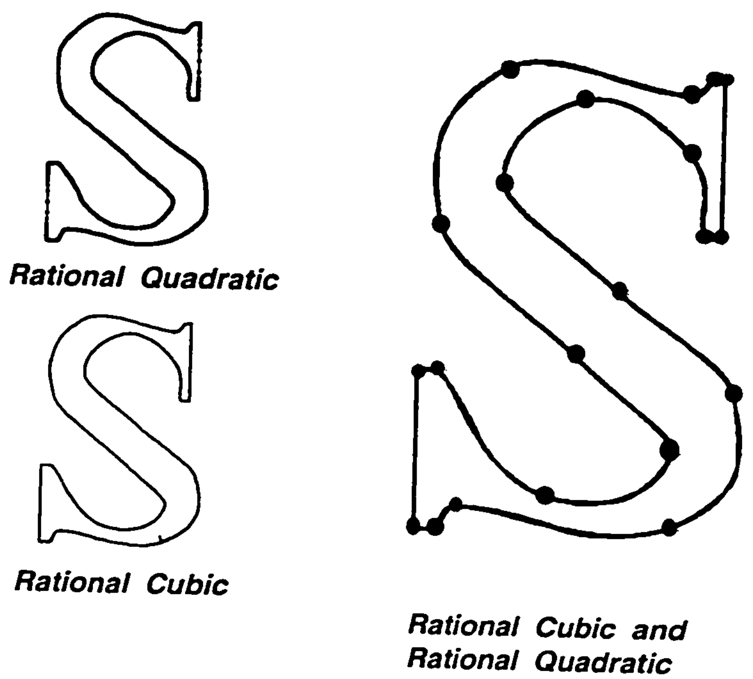


Figure 3.21: C^1 continuous rational cubic and rational quadratic : *an alphabet "S" is drawn*; where $r_i \geq$ *or* $\leq 3.0. \gamma_i \leq$ *or* ≥ 2.0 *i.e.*, ($\gamma_i = \frac{r_i}{2}$); uniform h_i's (=1) are taken.

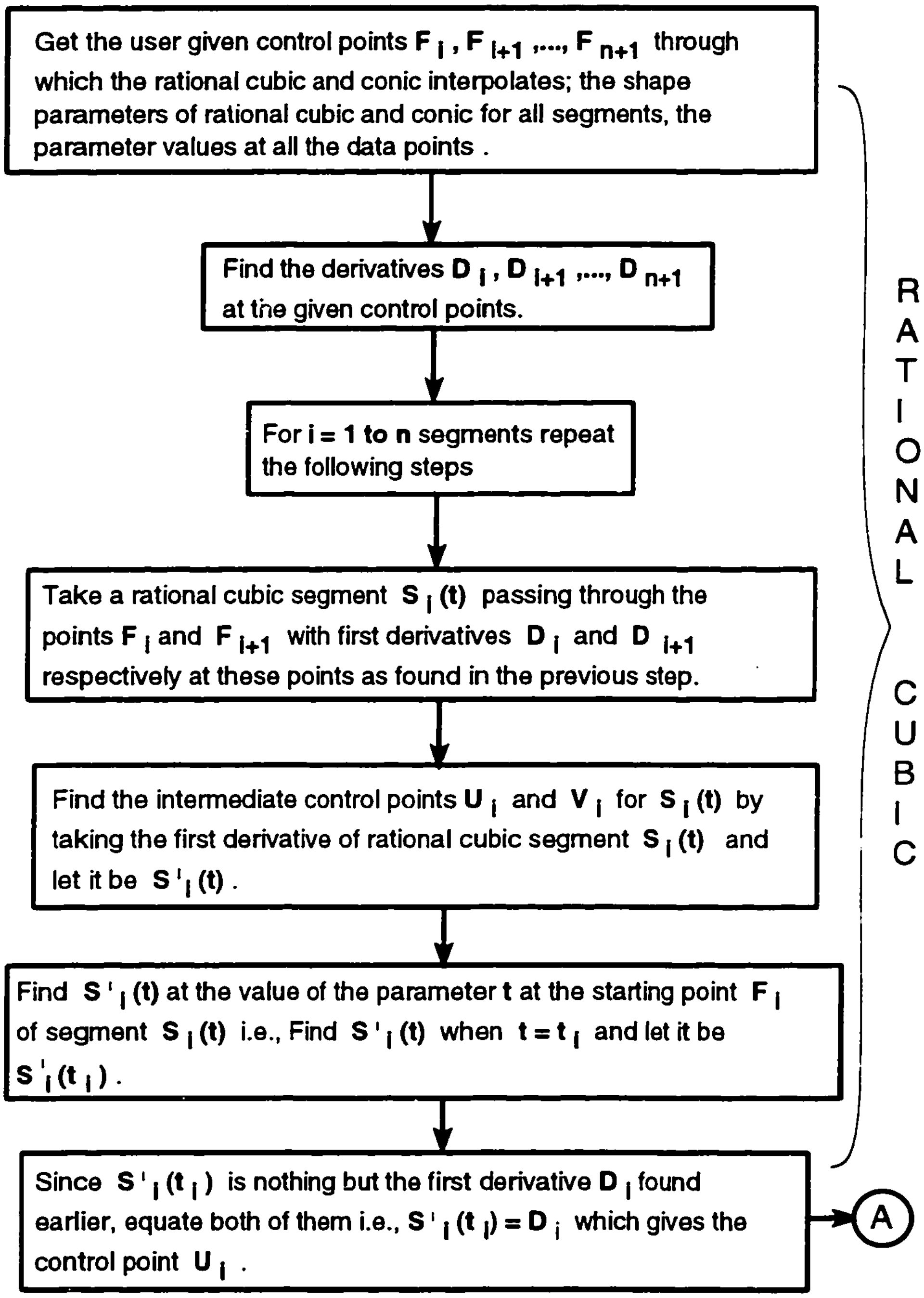


Figure 3.22: step-by-step procedure explaining the method of obtaining a C^1 continuous conic and a C^1 continuous rational cubic.

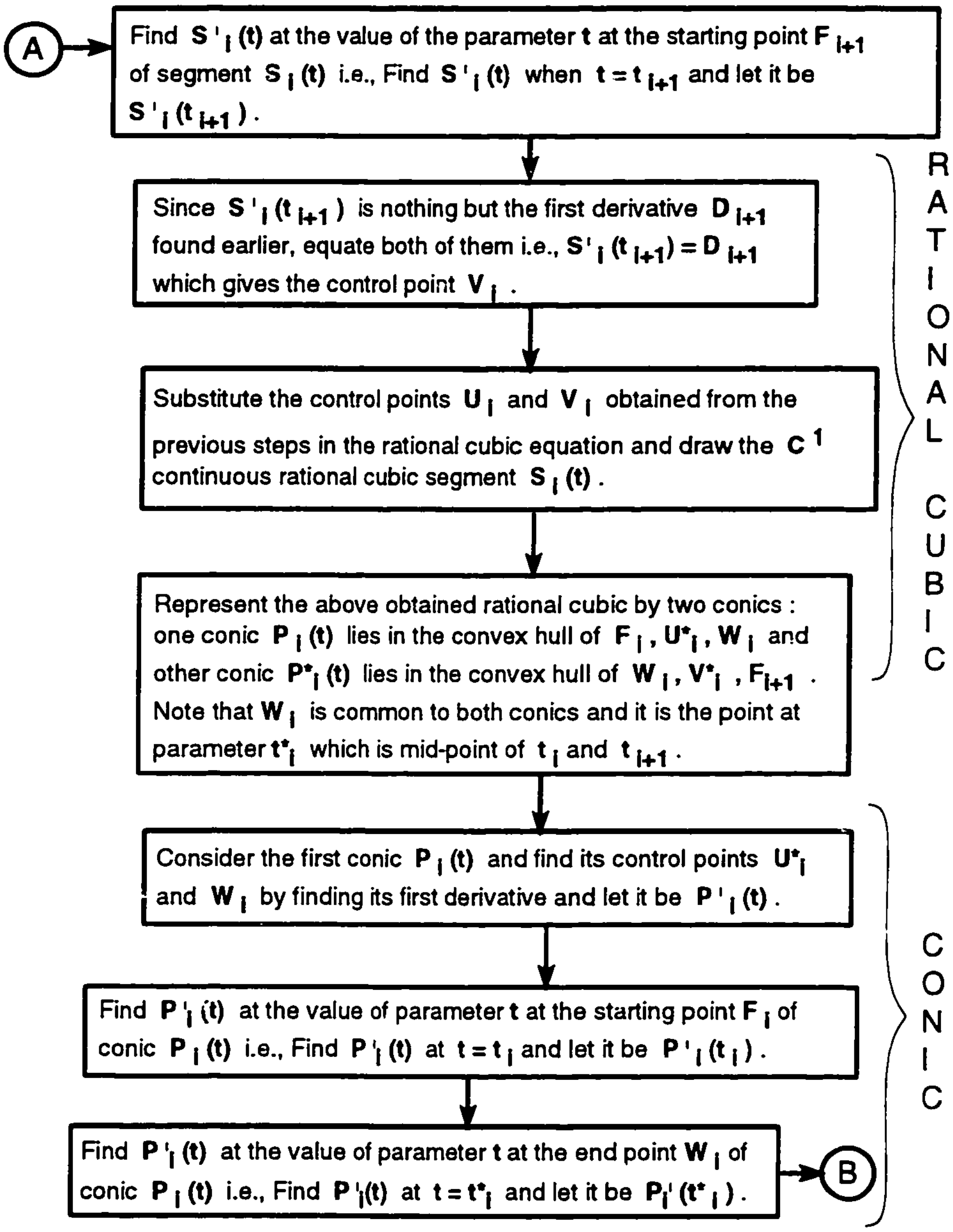


Figure 3.23: step-by-step procedure explaining the method of obtaining a C^1 continuous conic and a C^1 continuous rational cubic. (*cont...*)

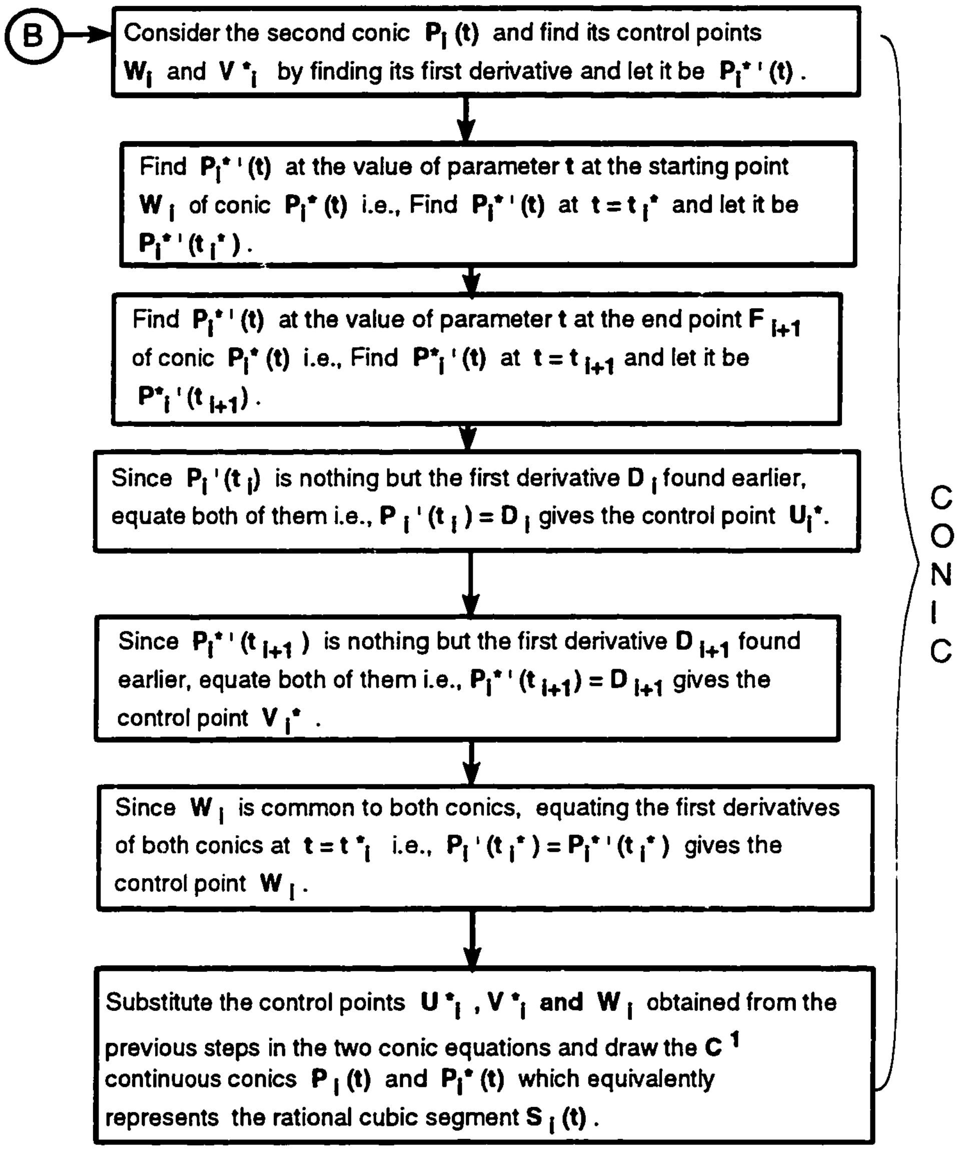


Figure 3.24: step-by-step procedure explaining the method of obtaining a C^1 continuous conic and a C^1 continuous rational cubic. (*cont...*)

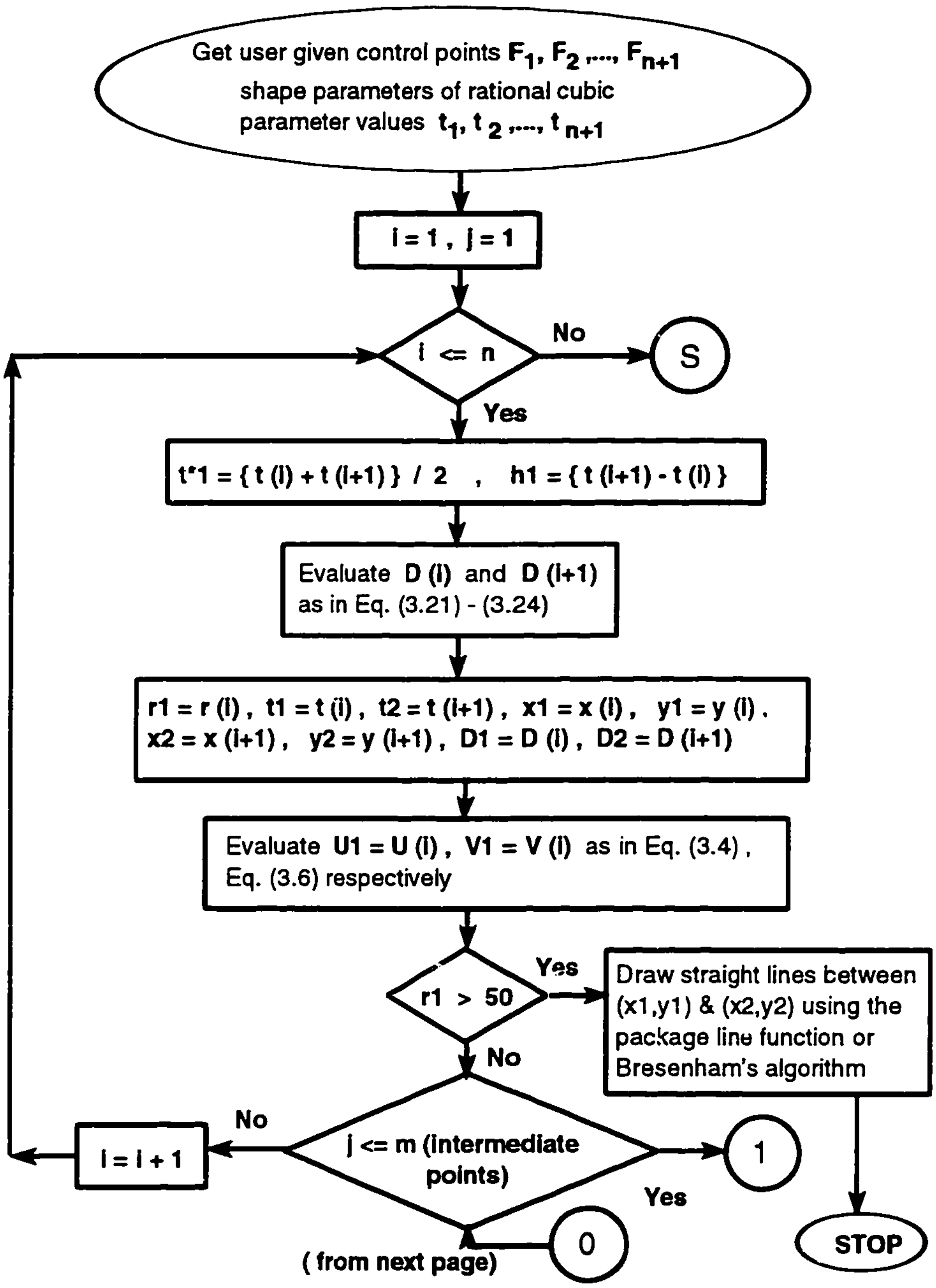


Figure 3.25: Flow-chart for obtaining a C^1 continuous rational cubic.

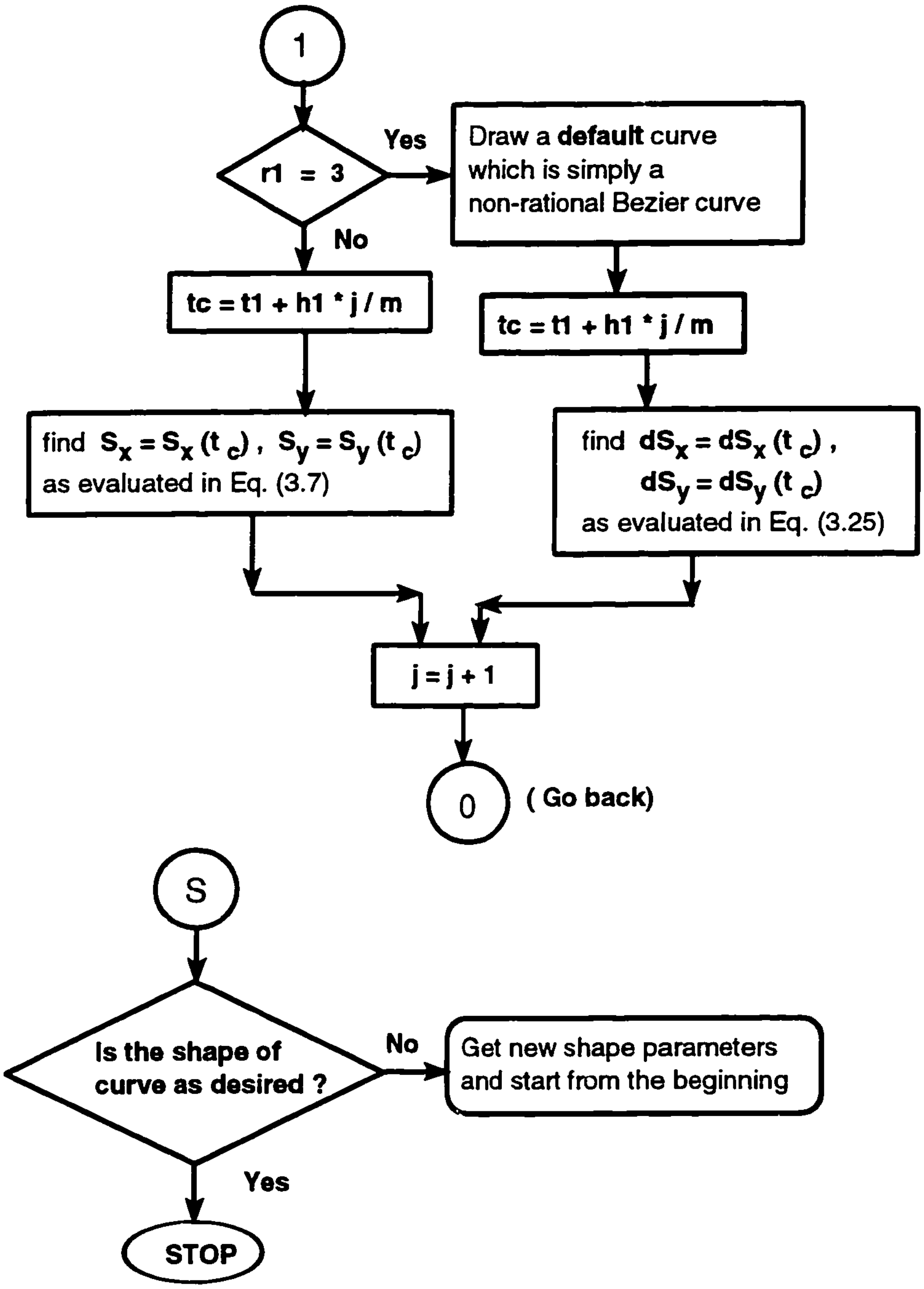


Figure 3.26: Flow-chart for obtaining a C^1 continuous rational cubic. (*cont...*)

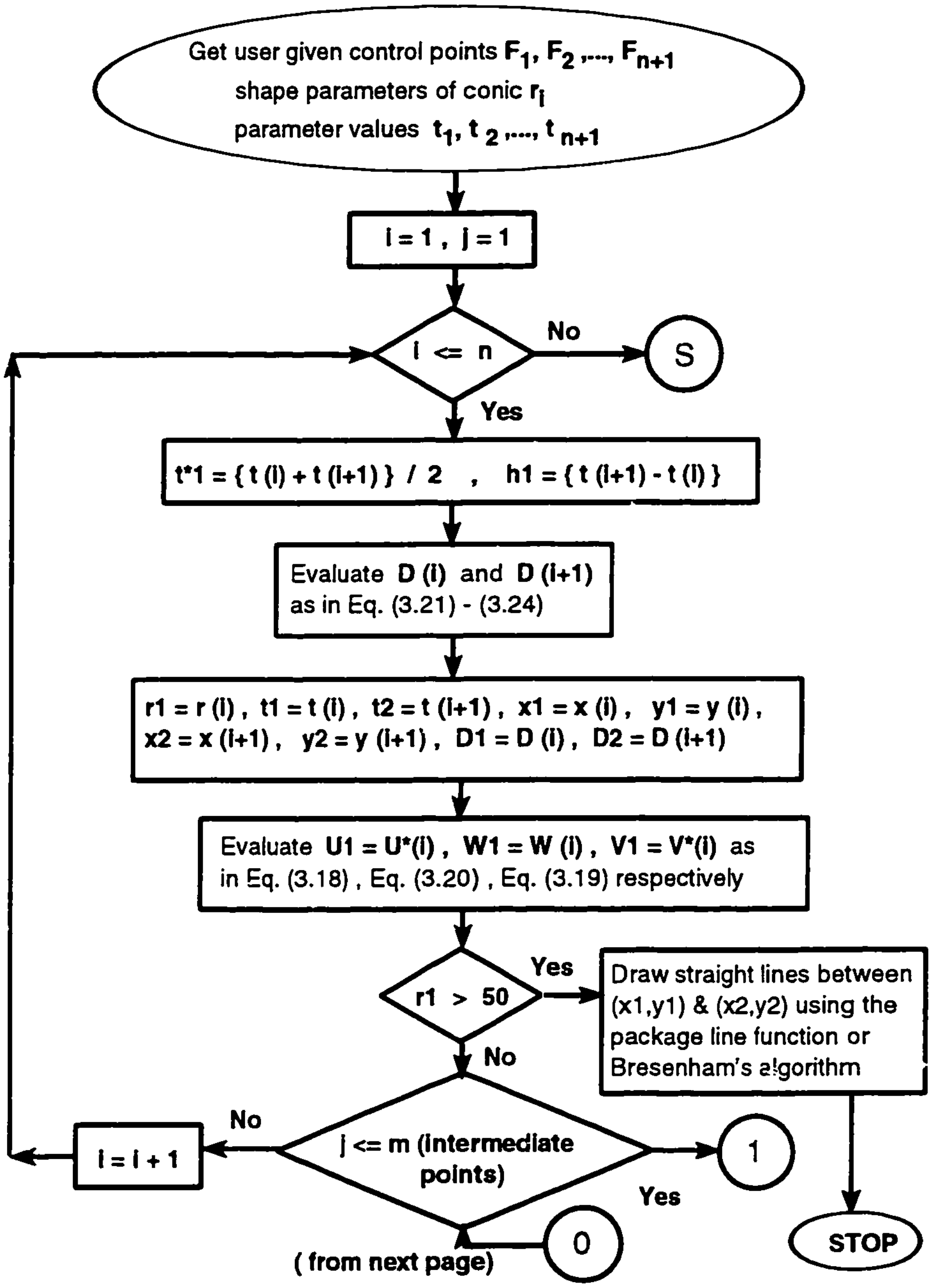


Figure 3.27: Flow-chart for obtaining a C^1 continuous conic.

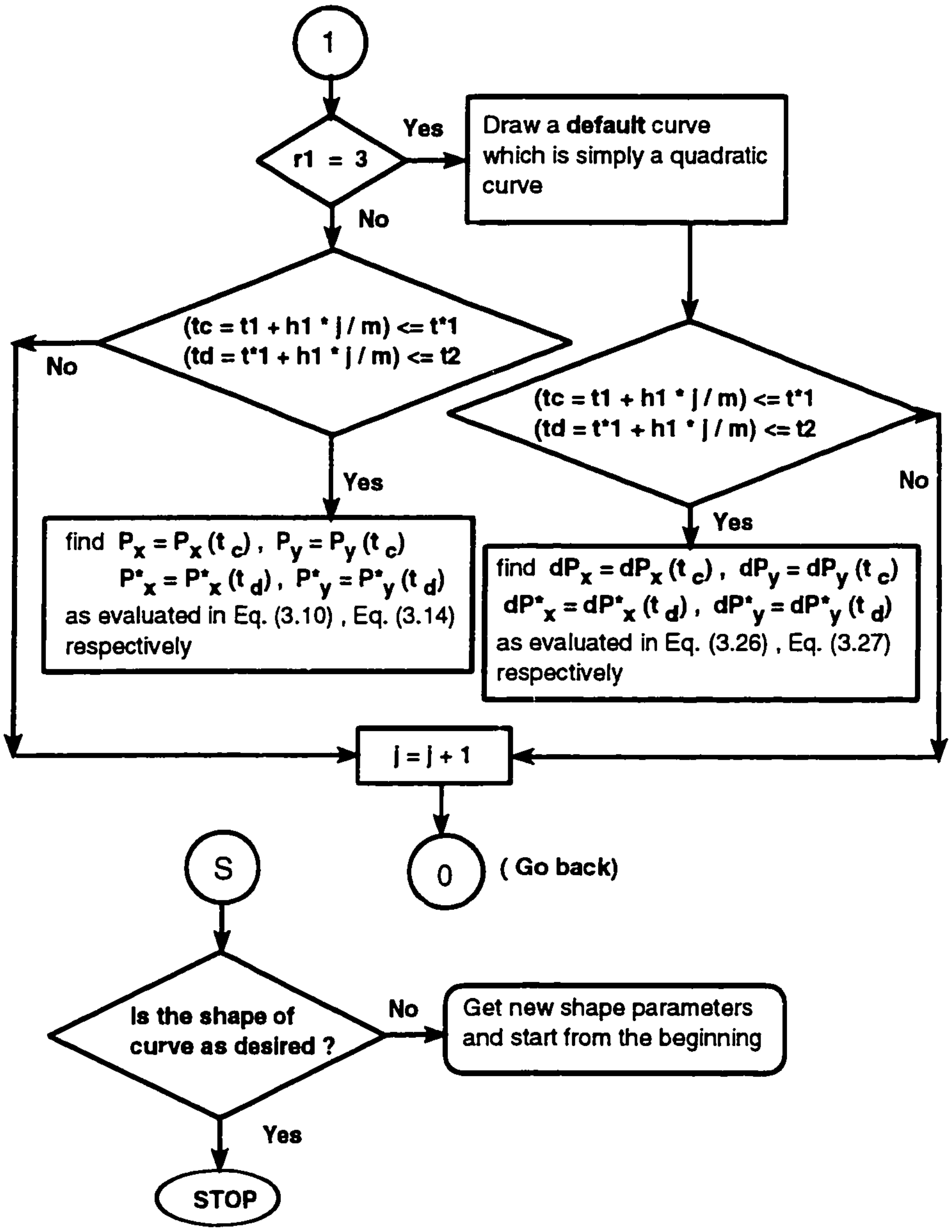


Figure 3.28: Flow-chart for obtaining a C^1 continuous conic. (*cont...*)

Chapter 4

Representation of a C^2 continuous rational cubic and its alternate C^2 continuous rational quadratic

In the previous chapter. a C^1 continuous rational cubic was represented by an alternate C^1 continuous rational quadratic. In this chapter, a C^2 continuous rational cubic will be alternatively represented by a C^2 continuous rational quadratic. All the description about rational cubic and conic - their equations, representations, notations ; is same as in the previous chapter on C^1 continuity.

Here, the following two cases of representing a C^2 continuous rational cubic by a C^2 continuous conic will be investigated

- Using the derivatives, obtained from the C^2 continuous rational cubic, in the equations of conic to get C^2 continuous conic.

- Finding the derivatives for C^2 continuous rational cubic and C^2 continuous conic separately and using them in the respective equations of rational cubic and conic.

However, the second case is considered as just a special case because it works properly only under certain conditions. The first case is the generally accepted case. Also, different types of representing a C^2 continuous rational cubic is discussed.

4.1 C^2 rational cubic interpolant

Let a piecewise rational cubic parametric function $S \in C^2[t_1, t_n]$ be, as defined by equation (4.1) of the previous chapter on C^1 continuity. For the rational cubic $S(t)$ to be C^2 continuous, the piecewise segment $S_i(t)$ passing through the points F_i and F_{i+1} and the next segment $S_{i+1}(t)$ passing through the end points $\hat{F}_{i+1}$ and $\hat{F}_{i+2}$, be such that the end points F_{i+1} and $\hat{F}_{i+1}$ should coincide, the first and second derivatives at the coinciding point of the two segments should be same.

Let $S_i'(t_i)$ and $S_i'(t_{i+1})$ be the first derivatives, $S_i''(t_i)$ and $S_i''(t_{i+1})$ be the second derivatives of the segment $S_i(t)$ at the endpoints F_i and F_{i+1} respectively. Similarly, let $S_{i+1}'(t_i)$ and $S_{i+1}'(t_{i+1})$ be the first derivatives, $S_{i+1}''(t_i)$ and $S_{i+1}''(t_{i+1})$ be the second derivatives of the segment $S_{i+1}(t)$ at the endpoints $\hat{F}_{i+1}$ and $\hat{F}_{i+2}$ respectively.

Then the conditions for C^2 continuous rational cubic are given as :

$$F_{i+1} = \hat{F}_{i+1} \quad , S_i'(t_{i+1}) = S_{i+1}'(t_i) \quad , S_i''(t_{i+1}) = S_{i+1}''(t_i)$$

In the previous chapter on C^1 continuous rational cubic, the equality of first derivatives of adjacent rational cubic segments has been discussed from which the control points U_i and V_i of the rational cubic were found. Now, consider the C^2 continuity by taking the second derivative of the rational cubic equation and applying the condition of $S_i''(t_{i+1}) = S_{i+1}''(t_i)$ to get the D_i's. These D_i's are then substituted in the rational cubic equation to get curvature continuous rational cubic.

Consider the *Hermite interpolation form* of the rational cubic which was found previously as,

$$S(t) = S_i(t) = \frac{F_i(1-\theta)^3 + (r_iF_i + h_iD_i)\theta(1-\theta)^2 + (r_iF_{i+1} - h_iD_{i+1})\theta^2(1-\theta) + F_{i+1}\theta^3}{(1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3} \tag{4.1}$$

The first derivative of $S_i(t)$ is given by

$$S_i'(t) = \frac{r_i(U_i - F_i)(1-\theta)^4 + (3(F_i + F_{i+1}) + r_i^2(V_i - U_i))\theta^2(1-\theta)^2}{h_i[\mathbf{D}]^2}$$

$$+\frac{2r_i(V_i - F_i)\theta(1-\theta)^3 + 2r_i(F_{i+1} - U_i)\theta^3(1-\theta) + r_i(F_{i+1} - V_i)\theta^4}{h_i[\mathbf{D}]^2} \tag{4.2}$$

where $\mathbf{D}$ is the denominator of equation (4.1).

OR

$$S_i'(t) = \frac{N}{D}$$

where N and D are numerator and denominator, respectively in equation (4.2).

The second derivative of $S_i(t)$, after much simplification, is then obtained as :

Let

$$B1 = (h_i((1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3)^2)$$

$$B2 = (4h_i(D_{i+1} - D_i) + 2(r_i^2 - 2r_i + 3)(F_{i+1} - F_i) - 2r_ih_i(D_{i+1} - D_i))\theta(1-\theta)^3$$

$$B3 = (4h_i(D_{i+1} - D_i) - 2(r_i^2 - 2r_i + 3)(F_{i+1} - F_i) - 2r_ih_i(D_{i+1} - D_i))\theta^3(1-\theta)$$

$$B4 = (2r_i(F_{i+1} - F_i) - 2h_i(D_{i+1} + 2D_i))(1-\theta)^4 + (6r_ih_iD_{i+1})\theta^2(1-\theta)^2$$

$$B5 = (2h_i(2D_{i+1} + D_i) - 2r_i(F_{i+1} - F_i))\theta^4$$

$$B6 = 2h_i((1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3)(r_i - 3)(1 - 2\theta)$$

Then

$$S_i''(t) = \frac{B1 \times (B2 + B3 + B4 + B5) - (N) \times B6}{h_i^3((1-\theta)^3 + r_i\theta(1-\theta)^2 + r_i\theta^2(1-\theta) + \theta^3)^4} \tag{4.3}$$

If $t = t_i$, θ *becomes* 0 and the equation (4.3) reduces to

$$S_i''(t_i) = \frac{2r_i(F_{i+1} - F_i) - 2h_i(D_{i+1} - D_i + r_iD_i)}{h_i^2} \tag{4.4}$$

Using the equation (4.4) for the segment $S_{i+1}(t)$, the equation for $S_{i+1}''(t_i)$ is then written as

$$S_{i+1}''(t_i) = \frac{2r_{i+1}(F_{i+2} - F_{i+1}) - 2h_{i+1}(D_{i+2} - D_{i+1} + r_{i+1}D_{i+1})}{h_{i+1}^2} \tag{4.5}$$

If $t = t_{i+1}$, θ *becomes* 1 and the equation (4.3) reduces to

$$S_i''(t_{i+1}) = \frac{2h_i(D_i - D_{i+1} + r_i D_{i+1}) - 2r_i(F_{i+1} - F_i)}{h_i^2} \tag{4.6}$$

Now, as mentioned earlier, for the piecewise rational cubic to be C^2 continuous it has to have *second-order continuity at the internal joints.* So, *equating the curvature at the end of the first segment $S_i''(t_{i+1})$ (equation (4.6)) with the curvature at the beginning of the next segment $S_{i+1}''(t_{i+1})$ (equation (4.5))* yields

$$\frac{h_{i+1}^2}{h_i}(D_i - D_{i+1} + r_i D_{i+1}) - \frac{r_i h_{i+1}^2}{h_i^2}(F_{i+1} - F_i)$$

$$= r_{i+1}(F_{i+2} - F_{i+1}) - h_{i+1}(r_{i+1} D_{i+1} + D_{i+2} - D_{i+1})$$

which after simplification results in

$$h_{i+1} D_i + (h_i(r_{i+1} - 1) + h_{i+1}(r_i - 1))D_{i+1} + h_i D_{i+2} = h_{i+1} r_i \Delta_i + h_i r_{i+1} \Delta_{i+1} \tag{4.7}$$

When $i = 1$

$$h_2 D_1 + (h_1(r_2 - 1) + h_2(r_1 - 1))D_2 + h_1 D_3 = h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2$$

When $i = 2$

$$h_3 D_2 + (h_2(r_3 - 1) + h_3(r_2 - 1))D_3 + h_2 D_4 = h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3$$

Applying equation (4.7) recursively over all the spline segments yields $n-2$ equations for the tangent vectors $D_k = S_k'(t)$, $2 \leq k \leq n-1$. In matrix form the result is

$$[M^*][S'] = [R]$$

Let

$$r_1 = r_1 - 1, \quad r_2 = r_2 - 1, \quad r_3 = r_3 - 1$$

$$r_1^* = r_{n-1} - 1, \quad r_2^* = r_{n-2} - 1$$

so that

$$M^* = \begin{bmatrix} h_2 & h_1(r_2) + h_2(r_1) & h_1 & 0 & \cdots \\ 0 & h_3 & h_2(r_3) + h_3(r_2) & h_2 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 0 & h_{n-1} & h_{n-2}(r_1^*) + h_{n-1}(r_2^*) & h_{n-2} \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \cdots \\ \cdots \\ D_n \end{bmatrix}$$

$$R = \begin{bmatrix} h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2 \\ h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3 \\ \cdots \\ \cdots \\ \cdots \\ h_{n-1} r_{n-2} \Delta_{n-2} + h_{n-2} r_{n-1} \Delta_{n-1} \end{bmatrix}$$

Since there are only $n-2$ equations for the n tangent vectors, $[M^*]$ is not square and thus cannot be inverted to obtain the solution for $[S']$; that is, the problem is indeterminate. By assuming that the end tangent vectors D_1 and D_n are known, the problem becomes determinant. The matrix formulation is now

$$[M][S'] = [R] \tag{4.8}$$

where

$$M = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots \\ h_2 & h_1(r_2) + h_2(r_1) & h_1 & 0 & \cdots \\ 0 & h_3 & h_2(r_3) + h_3(r_2) & h_2 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 0 & h_{n-1} & h_{n-2}(r_1^*) + h_{n-1}(r_2^*) & h_{n-2} \\ \cdots & \cdots & 0 & 0 & 1 \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \cdots \\ \cdots \\ D_n \end{bmatrix}$$

$$R = \begin{bmatrix} D_1 \\ h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2 \\ h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3 \\ \cdots \\ h_{n-1} r_{n-2} \Delta_{n-2} + h_{n-2} r_{n-1} \Delta_{n-1} \\ D_n \end{bmatrix}$$

where $[M]$ *is now square and invertible. Notice also that* $[M]$ *is tridiagonal, which reduces the computational work required to invert it. Also,* $[M]$ *is nonsingular, diagonally dominant and inversion yields a unique solution.*

Thus the solution for $[S']$ is

$$[S'] = [M]^{-1}[R] \tag{4.9}$$

Once the D_i's are known they can be substituted in the respective equations of rational cubic and conic to get the C^2 continuous piecewise rational cubic and its

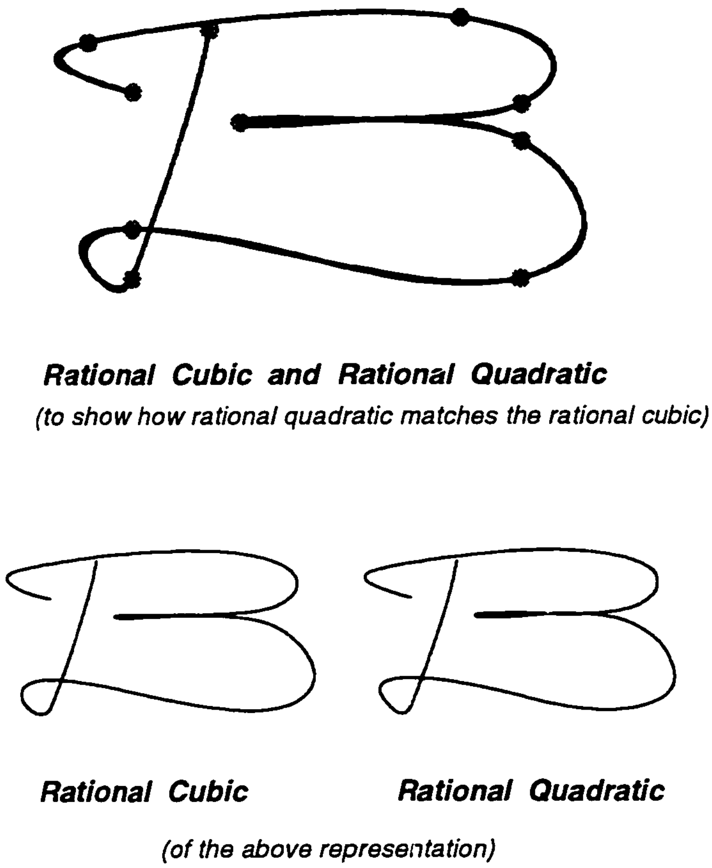


Figure 4.1: C^2 continuous rational cubic and rational quadratic : **Clamped end condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) are used with conics* with default values for shape parameters while drawing and uniform h_i's (=1).

equivalent piecewise rational quadratic representation. The control points of conic were already found in the previous chapter on C^1 continuity.

4.1.1 Relaxed end condition

In the previous discussions the tangent vectors D_1 and D_n at the ends of the piecewise cubic curve were assumed known. This boundary condition is called **Clamped end condition**. The unknown internal tangent vectors are given by inversion of the tridiagonal matrix $[M]$ as described in equation (4.9).

Alternate boundary conditions may be desired in some cases when only a few data points are known or if physical constraints require accurate control of the curve shape at the ends. One alternative is to specify the curvature at the ends of the spline curve. When the curvature is **zero**, a **relaxed or natural end condition** is obtained.

At the beginning of the first spline segment $S''(t)$ at $t = t_1$, $\theta = 0$

$$\frac{2r_1(F_2 - F_1) - 2h_1(D_2 - D_1 + r_1 D_1)}{h_1^2} = 0$$

$$r_1(F_2 - F_1) = h_1(D_2 - D_1 + r_1 D_1)$$

$$D_1(r_1 h_1 - h_1) + D_2 h_1 = r_1(F_2 - F_1)$$

At the end of the last spline segment $S''(t)$ at $t = t_n$, $\theta = 1$

$$\frac{2h_{n-1}(D_{n-1} - D_n + r_{n-1}D_n) - 2r_{n-1}(F_n - F_{n-1})}{h_{n-1}} = 0$$

$$(D_{n-1}h_{n-1} + D_n(r_{n-1}h_{n-1} - h_{n-1}) = r_{n-1}(F_n - F_{n-1})$$

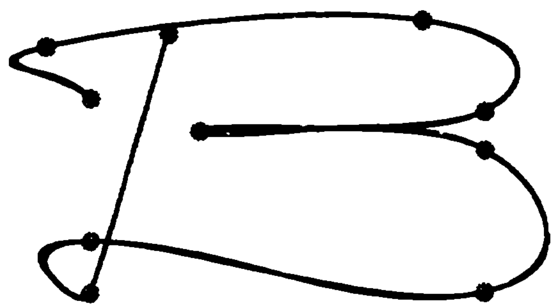
The matrix of equation (4.8) now becomes

$$[M][S'] = [R] \tag{4.10}$$

where

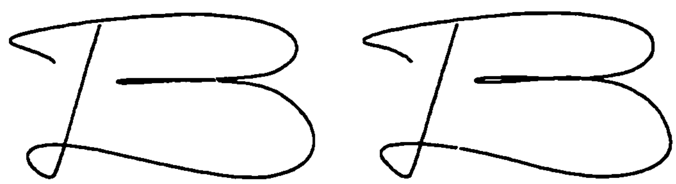
$$M = \begin{bmatrix} h_1 r_1 & h_1 & 0 & \dots & \dots \\ h_2 & h_1(r_2) + h_2(r_1) & h_1 & 0 & \dots \\ 0 & h_3 & h_2(r_3) + h_3(r_2) & h_2 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & h_{n-1} & h_{n-2}(r_1^*) + h_{n-1}(r_2^*) & h_{n-2} \\ \dots & \dots & 0 & h_{n-1} & h_{n-1}(r_1^*) \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \dots \\ \dots \\ D_n \end{bmatrix}$$



Rational Cubic and Rational Quadratic

(to show how rational quadratic matches the rational cubic)



Rational Cubic ***Rational Quadratic***

(of the above reprasentation)

Figure 4.2: C^2 continuous rational cubic and rational quadratic : **Relaxed end condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) are used with conics* with default values for shape parameters while drawing and uniform h_i's (=1).

$$R = \begin{bmatrix} r_1(F_2 - F_1) \\ h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2 \\ h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3 \\ \cdots \\ h_{n-1} r_{n-2} \Delta_{n-2} + h_{n-2} r_{n-1} \Delta_{n-1} \\ r_{n-1}(F_n - F_{n-1}) \end{bmatrix}$$

4.1.2 Normalized Condition

An alternate approximation for the t_i segment parameter values to that done till now (taking different values for t_i), is to *normalize them to unity.* Thus, $0 \leq t \leq 1$, so that $h_i = t_{i+1} - t_i = 1 - 0 = 1$ for all segments and $\theta(t) = \frac{t-0}{1} = t$.

The matrix formulation of equation (4.8) now becomes

$$[M][S'] = [R] \tag{4.11}$$

where

$$M = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots \\ 1 & r_2 + r_1 & 1 & 0 & \cdots \\ 0 & 1 & r_3 + r_2 & 1 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 0 & 1 & r_1^* + r_2^* & 1 \\ \cdots & \cdots & 0 & 0 & 1 \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \cdots \\ \cdots \\ D_n \end{bmatrix}$$

$$R = \begin{bmatrix} D_1 \\ r_1\Delta_1 + r_2\Delta_2 \\ r_2\Delta_2 + r_3\Delta_3 \\ \cdots \\ r_{n-2}\Delta_{n-2} + r_{n-1}\Delta_{n-1} \\ D_n \end{bmatrix}$$

4.1.3 Closed curve case

Now, consider the C^2 continuous closed curves which requires that the *first and last control points should coincide and both the tangent vector and the curvature at the two ends be equal i.e.,*

$$F_1 = F_n S_1'(t_1) = S_n'(t_n) \quad (or D_1 = D_n) S_1''(t_1) = S_n''(t_n)$$

From equation (4.3), when $t = t_1$, θ becomes 0 and $S_1''(t_1)$ is given by

$$S_1''(t_1) = \frac{2r_1(F_2 - F_1) - 2h_1(D_2 - D_1 + r_1 D_1)}{h_1^2} \tag{4.12}$$

From equation (4.3), when $t = t_n$, θ becomes 1 and $S''_n(t_n)$ is given by

$$S''_n(t_n) = \frac{2h_{n-1}(D_{n-1} - D_n + r_{n-1}D_n) - 2r_{n-1}(F_n - F_{n-1})}{h_{n-1}^2} \tag{4.13}$$

Recalling that $F_n = F_1$ and $D_n = D_1$, substituting them in the equation (4.13) gives

$$S''_n(t_n) = \frac{2h_{n-1}(D_{n-1} - D_1 + r_{n-1}D_1) - 2r_{n-1}(F_1 - F_{n-1})}{h_{n-1}^2} \tag{4.14}$$

Equating the equations (4.12) and (4.14) and simplifying, results in

$$D_1(h_1(r_{n-1} - 1) + h_{n-1}(r_1 - 1)) + D_2h_{n-1} + D_{n-1}h_1 \tag{4.15}$$

$$= \frac{h_{n-1}}{h_1}(r_1(F_2 - F_1)) + \frac{r_{n-1}}{h_{n-1}}(h_1(F_1 - F_{n-1}))$$

The tangent vectors at the internal joints are again obtained using equation (4.8). However, because all the tangent vectors are no longer independent (recall $D_1 = D_n$), [**M**] is now $(n-1) \times (n-1)$ with the first row given by the coefficients of equation (4.15), *i.e.*,

$$[M][S'] = [R] \tag{4.16}$$

where

$$M = \begin{bmatrix} h_1(r_1^*) + h_{n-1}(r_1) & h_{n-1} & 0 & \dots & h_1 \\ h_2 & h_1(r_2) + h_2(r_1) & h_1 & 0 & \dots \\ 0 & h_3 & h_2(r_3) + h_3(r_2) & h_2 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & h_{n-1} & h_{n-2}(r_1^*) + h_{n-1}(r_2^*) & h_{n-2} \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \dots \\ \dots \\ D_{n-1} \end{bmatrix}$$

$$R = \begin{bmatrix} \frac{h_{n-1}}{h_1}(r_1(F_2 - F_1)) + \frac{r_{n-1}}{h_{n-1}}(h_1(F_1 - F_{n-1})) \\ h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2 \\ h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3 \\ \dots \\ h_{n-1} r_{n-2} \Delta_{n-2} + h_{n-2} r_{n-1} \Delta_{n-1} \end{bmatrix}$$

Note that the matrix is no longer tridiagonal. The closed curve discussed so far is called the *cyclic closed curve.* Next. consider the *anticyclic closed curve* which is similar to the cyclic curve except that

$$S_1'(t_1) = -S_n'(t_n) \quad (or D_1 = -D_n) S_1''(t_1) = -S_n''(t_n) \tag{4.17}$$

Following the same procedure used for the cyclic end condition, yields

$$D_1(h_1(1 - r_{n-1}) + h_{n-1}(1 - r_1)) - D_2 h_{n-1} + D_{n-1} h_1 \tag{4.18}$$

$$= \frac{r_{n-1}}{h_{n-1}}(h_1(F_1 - F_{n-1})) - \frac{h_{n-1}}{h_1}(r_1(F_2 - F_1))$$

Equation (4.17) shows that the only effect of imposing the anticyclic end condition is to change the sign of the $M(1,1), M(1,2)$ in the **[M]** matrix and to change the sign of the second term at $R(1,1)$ in equation (4.16).

The matrix form can be written as

$$[M][S'] = [R] \tag{4.19}$$

Let

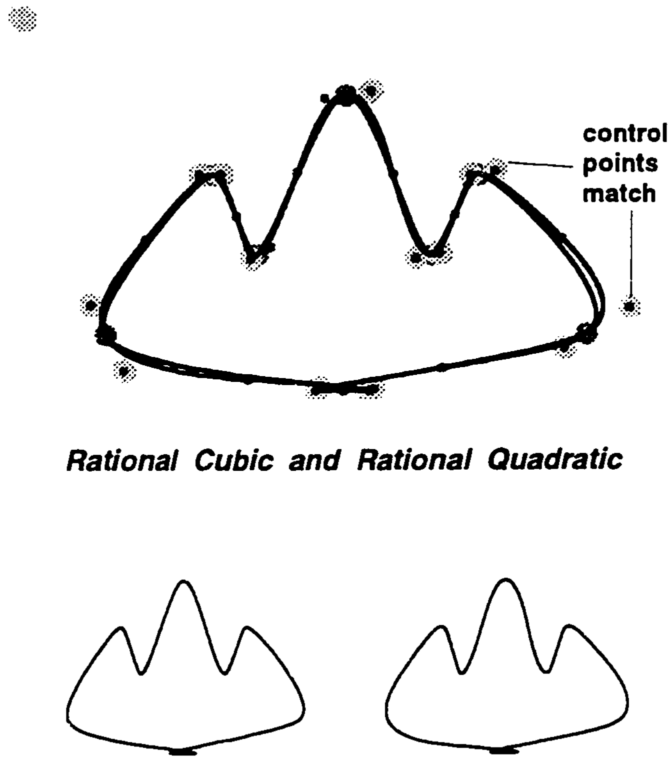
$$Y1 = h_1(-r_1^*) + h_{n-1}(-r_1), \qquad Y2 = h_{n-2}(r_1^*) + h_{n-1}(r_2^*)$$

so that

$$M = \begin{bmatrix} Y1 & -h_{n-1} & 0 & \dots & h_1 \\ h_2 & h_1(r_2) + h_2(r_1) & h_1 & 0 & \dots \\ 0 & h_3 & h_2(r_3) + h_3(r_2) & h_2 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & h_{n-1} & Y2 & h_{n-2} \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \dots \\ \dots \\ D_{n-1} \end{bmatrix}$$

$$R = \begin{bmatrix} \frac{r_{n-1}}{h_{n-1}}(h_1(F_1 - F_{n-1})) - \frac{h_{n-1}}{h_1}(r_1(F_2 - F_1)) \\ h_2 r_1 \Delta_1 + h_1 r_2 \Delta_2 \\ h_3 r_2 \Delta_2 + h_2 r_3 \Delta_3 \\ \cdots \\ h_{n-1} r_{n-2} \Delta_{n-2} + h_{n-2} r_{n-1} \Delta_{n-1} \end{bmatrix}$$



Rational Cubic and Rational Quadratic

Rational Cubic ***Rational Quadratic***

(of the above representation)

Figure 4.3: C^2 continuous rational cubic and rational quadratic : **Cyclic closed condition with control points matching case** *Derivatives of rational cubic (obtained by taking default shape parameter) are used with conics* with shape parameters $r > 3$ and $\gamma > 2$ and $\gamma = \frac{r}{2}$ while drawing and uniform h_i's $(=1)$.

4.2 C^2 rational quadratic interpolant

In the previous cases considered so far, the D_i's obtained from the C^2 continuity constraints of rational cubic were used in conics. Now, consider the case where the D_i's obtained by applying the C^2 continuity constraints on conic, are used in the conics *i.e.*, here the D_i's obtained from rational cubic C^2 continuity are used in rational cubic and D_i's obtained from the conic C^2 continuity are used in the conics. For this, we need to find the D_i's at the junction points of the C^2 continuous conics when D_1 and D_n are given. First, the second derivatives of the two conics are to be found at the parameter values $t = t_i, t_i^*, t_{i+1}$.

The second derivative of the first conic $P_i(t)$ at $t = t_i$ is

$$P_i''(t_i) = \frac{8}{h_i^2}(\gamma_i U_i^* - \gamma_i F_i - \gamma_i^2 U_i^* + \gamma_i^2 F_i - F_i + W_i)$$

The second derivative of the first conic $P_i(t)$ at $t = t_i^*$ is

$$P_i''(t_i^*) = \frac{8}{h_i^2}(\gamma_i U_i^* - \gamma_i W_i - \gamma_i^2 U_i^* + \gamma_i^2 W_i + F_i - W_i)$$

The second derivative of the second conic $P_i^*(t)$ at $t = t_i^*$ is

$$P_i^{*''}(t_i^*) = \frac{8}{h_i^2}(\gamma_i V_i^* - \gamma_i W_i + \gamma_i^2 W_i - \gamma_i^2 V_i^* + F_{i+1} - W_i)$$

The second derivative of the second conic $P_i^*(t)$ at $t = t_{i+1}$ is

$$P_i^{*''}(t_{i+1}) = \frac{8}{h_i^2}(\gamma_i V_i^* - \gamma_i F_{i+1} - \gamma_i^2 V_i^* + \gamma_i^2 F_{i+1} - F_{i+1} + W_i)$$

Then the following conditions are checked for the C^2 continuity of conics

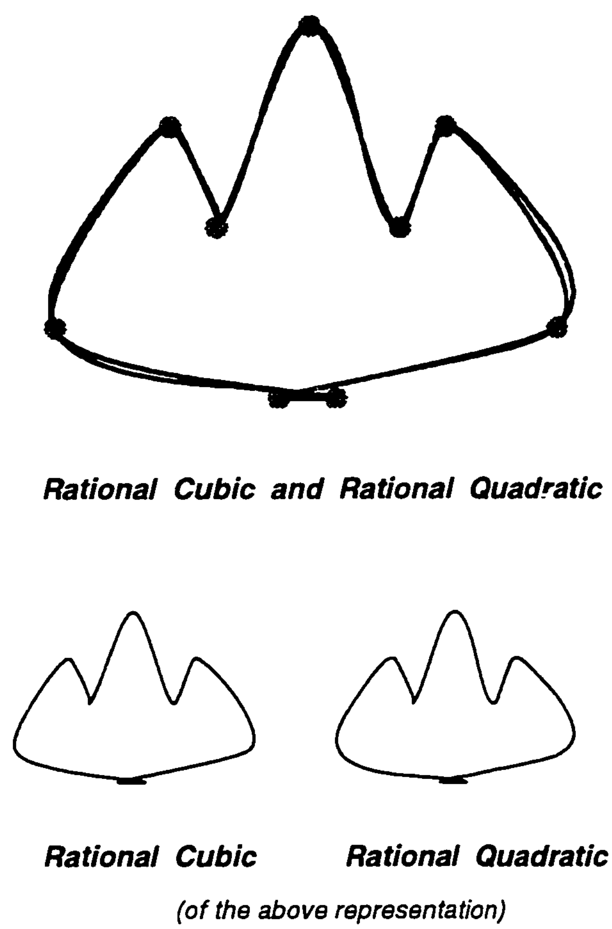


Figure 4.4: C^2 continuous rational cubic and rational quadratic : **Anticyclic closed condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) are used with conics* with shape parameters $r > 3$ and $\gamma > 2$ and $\gamma = \frac{r}{2}$ while drawing and uniform h_i's $(=1)$.

- The second derivative of the first conic at $t = t_i^*$ should be equal to the second derivative of the second conic at $t = t_i^*$, that is

$$P_i''(t_i^*) = P_i^{*''}(t_i^*)$$

which results in

$$(\gamma_i - \gamma_i^2)U_i^* + F_i = (\gamma_i - \gamma_i^2)V_i^* + F_{i+1}$$

substituting the values of U_i^* and V_i^* found in the previous chapter on C^1 continuity and simplifying, we get

$$D_{i+1} = \frac{2(F_{i+1} - F_i)(\gamma_i - \gamma_i^2 + 1)}{h_i(1 - \gamma_i)} - D_i \tag{4.20}$$

- The second derivative of the first conic of the next $(i+1)$ segment $S_{i+1}(t)$ at $t = t_i$ should be equal to the second derivative of the second conic of the ith segment $S_i(t)$ at $t = t_{i+1}$, that is

$$P_{i+1}''(t_i) = P_i^{*''}(t_{i+1})$$

which results in

$$\frac{h_{i+1}D_{i+1}}{2}(1 - \gamma_{i+1}) = \frac{h_{i+1}^2}{h_i^2}(\frac{h_i D_{i+1}}{2}(\gamma_i - 1) + W_i - F_{i+1}) + F_{i+1} - W_{i+1}$$

substituting the values of W_i and W_{i+1} found in the previous chapter on C^1 continuity and simplifying, we get

$$\frac{h_{i+1}}{4\gamma_{i+1}}D_{i+2} + (\frac{h_{i+1}^2}{2h_i}(\gamma_i - 1) - \frac{h_{i+1}^2}{4\gamma_i h_i} - \frac{h_{i+1}}{4\gamma_{i+1}} - \frac{h_{i+1}}{2}(1 - \gamma i + 1))D_{i+1} + \frac{h_{i+1}^2}{4\gamma_i h_i}D_i$$

$$= \frac{F_{i+2}}{2} - \frac{F_{i+1}}{2} - \frac{h_{i+1}^2}{h_i}(\frac{F_i - Fi + 1}{2}) \tag{4.21}$$

when $i = 1$, the equation (4.20) becomes

$$\frac{h_2}{4\gamma_2}D_3 + (\frac{h_2^2}{2h_1}(\gamma_1 - 1) - \frac{h_2^2}{4\gamma_1 h_1} - \frac{h_2}{4\gamma_2} - \frac{h_2}{2}(1 - \gamma_2))D_2 + \frac{h_2^2}{4\gamma_1 h_1}D_1$$

$$= \frac{F_3}{2} - \frac{F_2}{2} - \frac{h_2^2}{h_1}(\frac{F_1 - F_2}{2})$$

when $i = 2$, the equation (4.20) becomes

$$\frac{h_3}{4\gamma_3}D_4 + (\frac{h_3^2}{2h_2}(\gamma_2 - 1) - \frac{h_3^2}{4\gamma_2 h_2} - \frac{h_3}{4\gamma_3} - \frac{h_3}{2}(1 - \gamma_3))D_3 + \frac{h_3^2}{4\gamma_2 h_2}D_2$$

$$= \frac{F_4}{2} - \frac{F_3}{2} - \frac{h_3^2}{h_2}(\frac{F_2 - F_3}{2})$$

the equation (4.20) can be converted into matrix form to obtain D_i, $i = 2, 3, \ldots, n-1$ when D_1 and D_n are given, and can be written as

$$[M][S'] = [R] \tag{4.22}$$

Let

$$X1 = \frac{h_2^2}{2h_1}(\gamma_1 - 1) - \frac{h_2^2}{4\gamma_1 h_1} - \frac{h_2}{4\gamma_2} - \frac{h_2}{2}(1 - \gamma_2)$$

$$X2 = \frac{h_3^2}{2h_2}(\gamma_2 - 1) - \frac{h_3^2}{4\gamma_2 h_2} - \frac{h_3}{4\gamma_3} - \frac{h_3}{2}(1 - \gamma_3)$$

$$X3 = \frac{h_{n-1}^2}{2h_{n-2}}(\gamma_{n-2} - 1) - \frac{h_{n-1}^2}{4\gamma_{n-2} h_{n-2}} - \frac{h_{n-1}}{4\gamma_{n-1}} - \frac{h_{n-1}}{2}(1 - \gamma_{n-1})$$

$$M = \begin{bmatrix} 1 & 0 & 0 & \dots & \dots \\ \frac{h_2^2}{4\gamma_1 h_1} & X1 & \frac{h_2}{4\gamma_2} & 0 & \dots \\ 0 & \frac{h_3^2}{4\gamma_2 h_2} & X2 & \frac{h_3}{4\gamma_3} & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & \frac{h_{n-1}^2}{4\gamma_{n-2} h_{n-2}} & X3 & \frac{h_{n-1}}{4\gamma_{n-1}} \\ \dots & \dots & 0 & 0 & 1 \end{bmatrix}$$

$$S' = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \dots \\ \dots \\ D_n \end{bmatrix}$$

$$R = \begin{bmatrix} D_1 \\ \frac{F_3}{2} - \frac{F_2}{2} - \frac{h_2^2}{h_1}(\frac{F_1 - F_2}{2}) \\ \frac{F_1}{2} - \frac{F_3}{2} - \frac{h_3^2}{h_2}(\frac{F_2 - F_3}{2}) \\ \dots \\ \frac{F_n}{2} - \frac{F_{n-1}}{2} - \frac{h_{n-1}^2}{h_{n-2}}(\frac{F_{n-2} - F_{n-1}}{2}) \\ D_n \end{bmatrix}$$

Thus the solution for $[S']$ is

$$[S'] = [M]^{-1}[R] \tag{4.23}$$

Once the D_i's are known they can be substituted in equations of rational quadratics to get the curvature continous piecewise rational quadratics. But here the continuity is between the conics of different segments of the cubic.

For the conics to be C^2 continuous internally within the same segment of the cubic which the conics are representing, the D_{i+1} in terms of D_i, that is equation (4.19) should be submitted in equation (4.20) which then becomes

Since

$$D_{i+2} = \frac{2(F_{i+2} - F_{i+1})(r_{i+1} - r_{i+1}^2 + 1)}{h_{i+1}(1 - r_{i+1})} - D_{i+1}$$

we get

$$\left(\frac{h_{i+1}^2}{2h_i}(\gamma_i - 1) - \frac{h_{i+1}^2}{4\gamma_i h_i} - \frac{h_{i+1}}{4\gamma_{i+1}} - \frac{h_{i+1}}{2}(1 - \gamma_{i+1}) - \frac{h_{i+1}}{4\gamma_{i+1}}\right)D_{i+1}$$

$$= \frac{F_{i+2}}{2} - \frac{F_{i+1}}{2} - \frac{h_{i+1}^2}{h_i}\left(\frac{F_i - F_{i+1}}{2}\right) - \frac{h_{i+1}}{4\gamma_{i+1}}\left(\frac{2(F_{i+2} - F_{i+1})(\gamma_{i+1} - \gamma_{i+1}^2 + 1)}{h_{i+1}(1 - \gamma_{i+1})}\right) - \frac{h_{i+1}^2}{4\gamma_i h_i}D_i$$

which after simplification results in the following equation

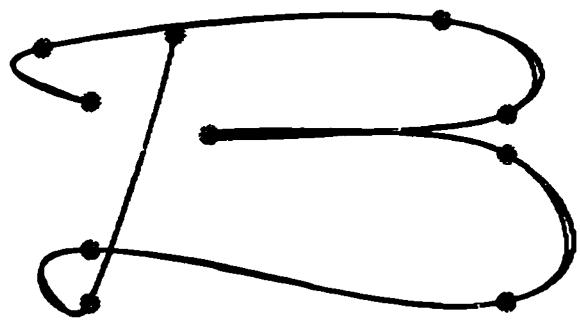
$$\left(\frac{h_{i+1}}{h_i}(\gamma_i - 1) - \frac{h_{i+1}}{2\gamma_i h_i} - \frac{1}{\gamma_{i+1}} - 1 + \gamma_{i+1}\right)D_{i+1}$$

$$= \frac{h_{i+1}}{h_i}(F_{i+1} - F_i) - \frac{F_{i+2} - F_{i+1}}{\gamma_{i+1}h_{i+1}(1 - \gamma_{i+1})} - \frac{h_{i+1}}{2\gamma_i h_i}D_i \qquad (4.24)$$

From the equation (4.23) the D_i's obtained are then used in the rational conic equations to get the C^2 continous rational conics which are approximating the C^2 continuous rational cubic whose D_i's are obtained from their C^2 continuity conditions.

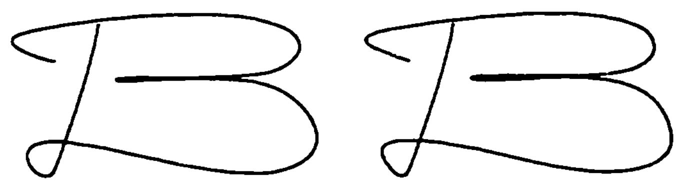
However, this case of different D_i's for rational cubic and conic works only when the h_i's are *uniform*, unlike the previous cases which works in the general manner for uniform or non-uniform h_i's.

All the properties that were discussed for C^1 rational cubic and conic - the convex hull property, the variation diminishing property, the Global tension property, the interval tension property, the looser curve, the corner effect; also apply to the C^2 rational cubic and conic as demonstrated in the figures later. Since, by applying the C^2 continuity conditions only the D_i's are found whereas the control points are found from the C^1 continuity case only, the control points matching case also applies to the C^2 continuous rational cubic and conic. Also, the default and best cases discussed in the C^1 continuity case applies for the C^2 continuous rational cubic and conic as demonstrated in the figures.



Rational Cubic and Rational Quadratic

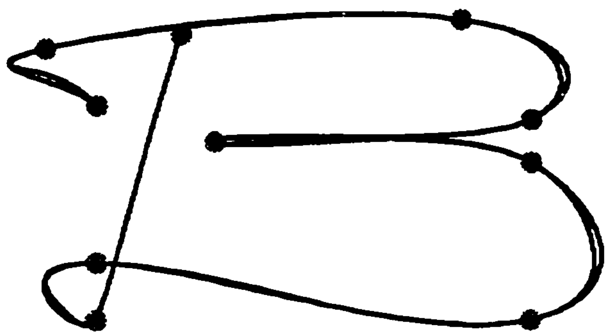
(to show how rational quadratic matches the rational cubic)



Rational Cubic ***Rational Quadratic***

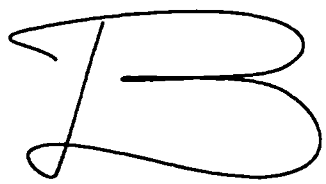
(of the above representation)

Figure 4.5: C^2 continuous rational cubic and rational quadratic : **Clamped end condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) and rational quadratic (obtained by taking default shape parameter) are used separately* with default values for shape parameters and uniform h_i's $(=1)$.

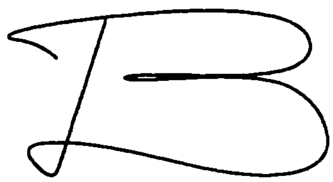


Rational Cubic and Rational Quadratic

(to show how rational quadratic matches the rational cubic)



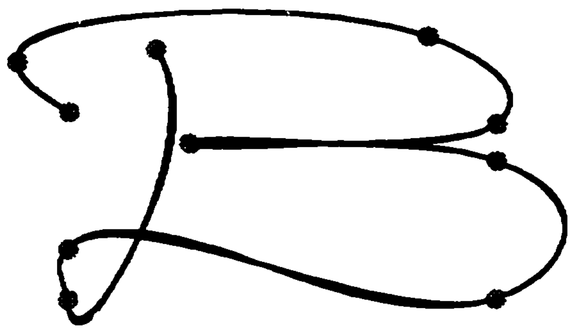
Rational Cubic



Rational Quadratic

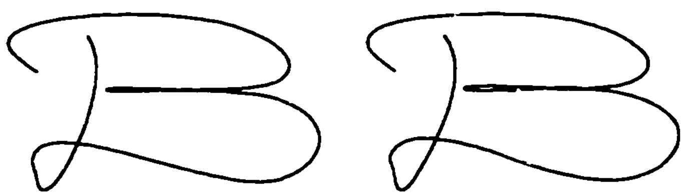
(of the above representation)

Figure 4.6: C^2 continuous rational cubic and rational quadratic : **Relaxed end condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) and rational quadratic (obtained by taking default shape parameter) are used separately* with default values for shape parameters and uniform h_i's $(=1)$.



Rational Cubic and Rational Quadratic

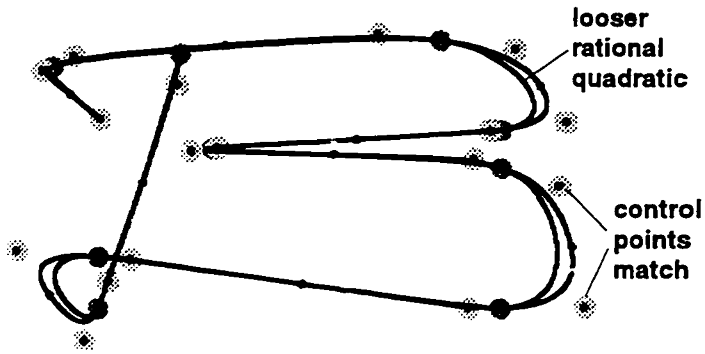
(to show how rational quadratic matches the rational cubic)



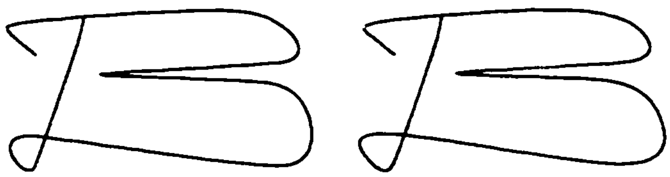
Rational Cubic ***Rational Quadratic***

(of the above representation)

Figure 4.7: C^2 continuous rational cubic and rational quadratic : **Clamped end condition case** *Derivatives of rational cubic (obtained by taking default shape parameter) are used with conics* with default values for shape parameters while drawing and chord length for h_i's.



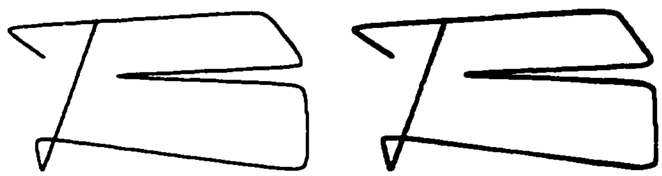
Rational Cubic and Rational Quadratic



Rational Cubic *Rational Quadratic*

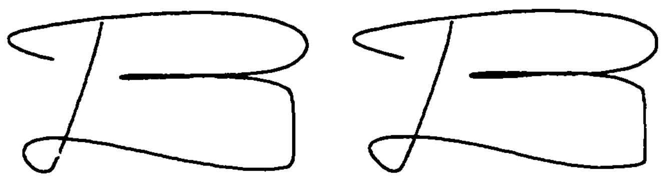
(of the above representation)

Figure 4.8: C^2 continuous rational cubic and rational quadratic : **Clamped end condition with control points matching case** *Derivatives of rational cubic (obtained by taking shape parameter $r > 3$) are used with conics* with shape parameters $r > 3$ and $\gamma > 2$ and $\gamma = \frac{r}{2}$ while drawing and uniform h_i's (=1).



Rational Cubic **Rational Quadratic**

(a)



Rational Cubic **Rational Quadratic**

(b)

Figure 4.9: C^2 continuous rational cubic and rational quadratic : (a) *Global tension property*; (b) *Interval tension property* ; uniform h_i's (=1) are taken for both figures.

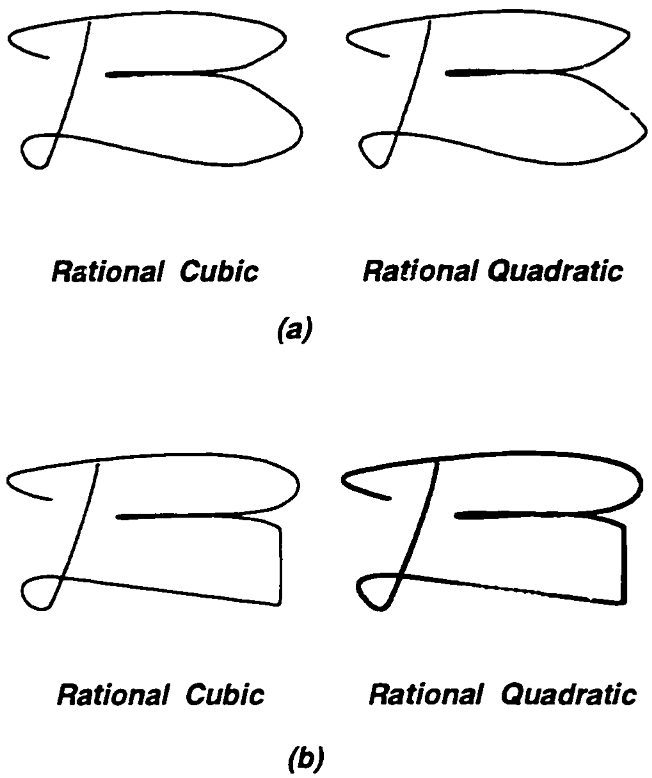


Figure 4.10: C^2 continuous rational cubic and rational quadratic : (a) *Looser curve case* ; (b) *Point tension property*; uniform h_i's $(=1)$ are taken for both figures.

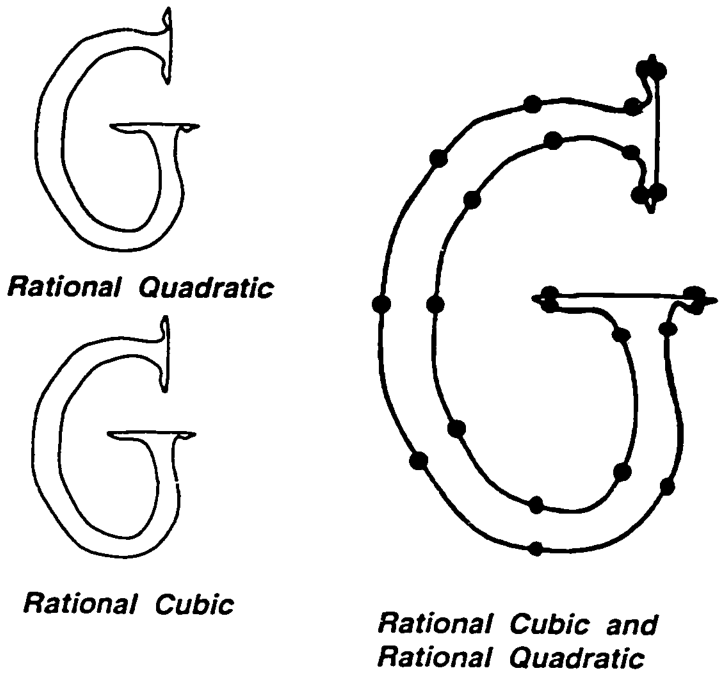


Figure 4.11: C^2 continuous rational cubic and rational quadratic : *an alphabet "G" is drawn*; default values for shape parameters are taken.

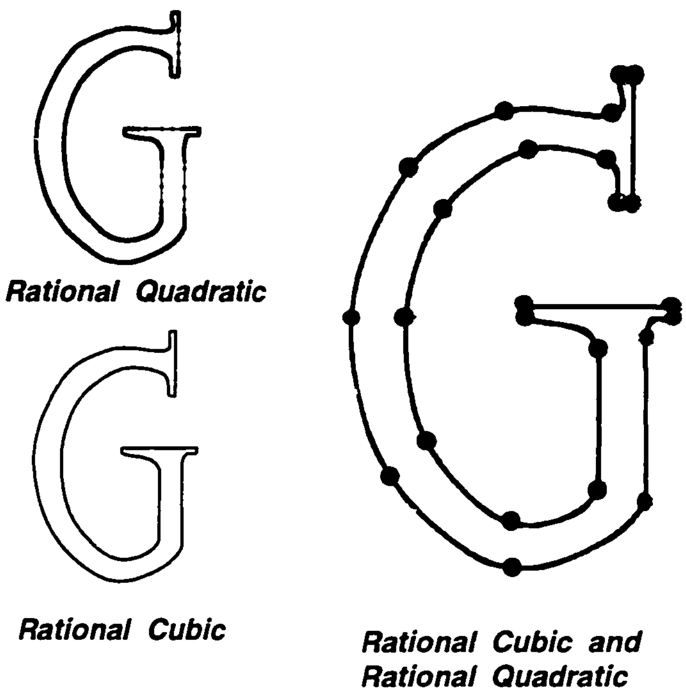


Figure 4.12: C^2 continuous rational cubic and rational quadratic : *an alphabet "G" is drawn*; where $r_i \geq 3.0, \gamma_i \geq 2.0$ *i.e.*, ($\gamma_i = \frac{r_i}{2}$); uniform h_i's (=1) are taken.

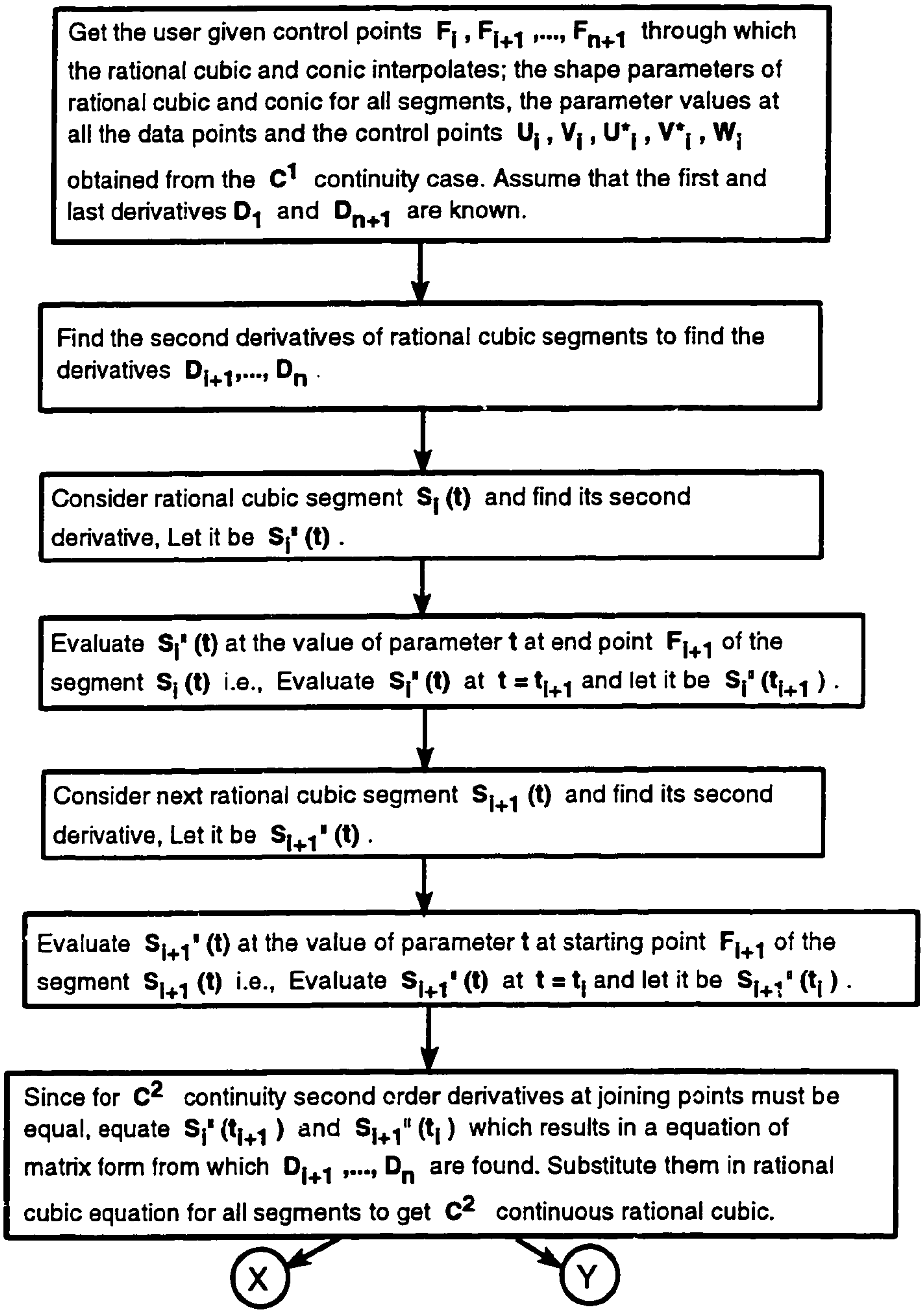


Figure 4.13: step-by-step procedure explaining the method of obtaining a C^2 continuous conic and a C^2 continuous rational cubic.

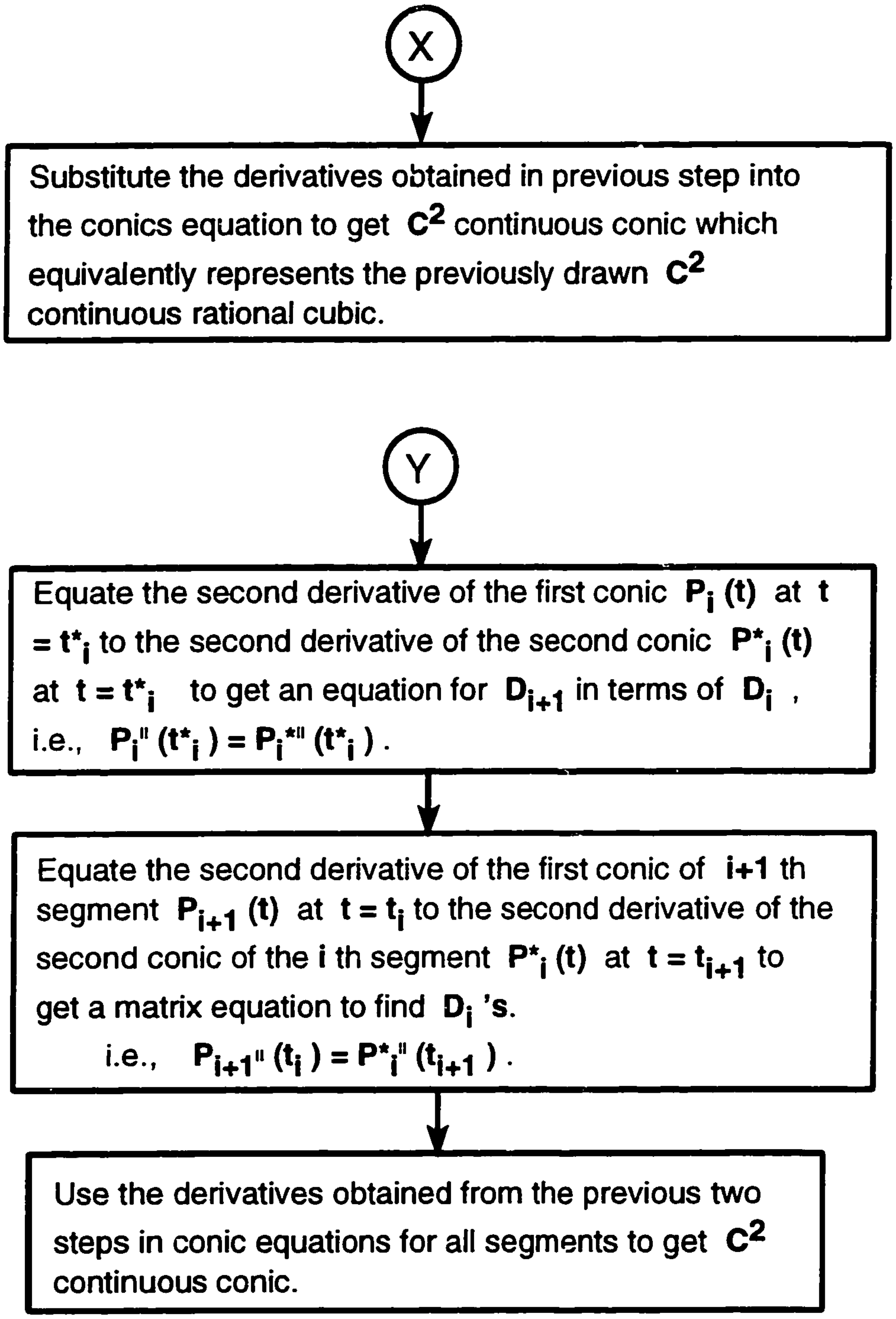


Figure 4.14: step-by-step procedure explaining the method of obtaining a C^2 continuous conic and a C^2 continuous rational cubic. (*cont...*)

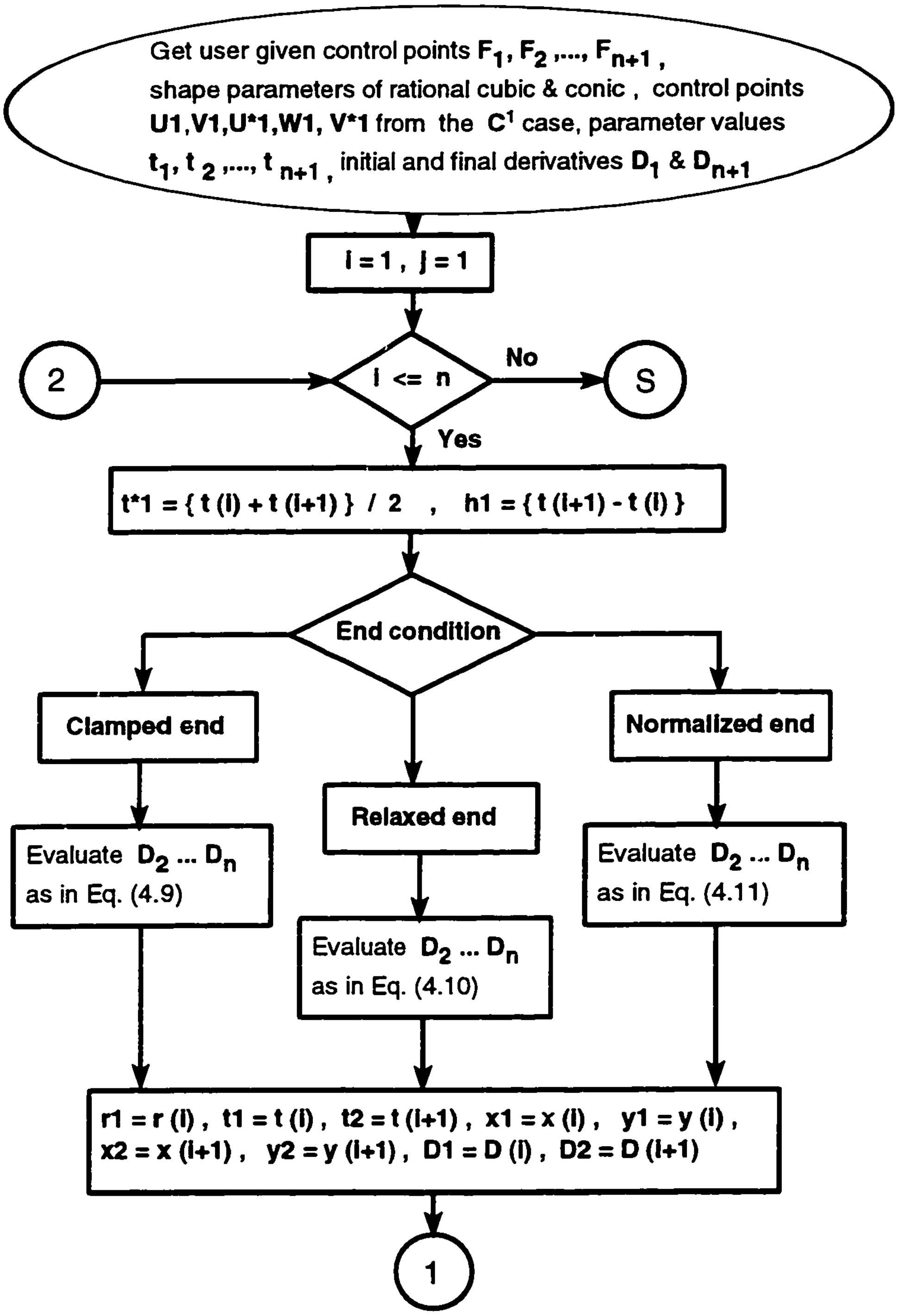


Figure 4.15: Flow-chart for obtaining a C^2 continuous rational cubic and conic.

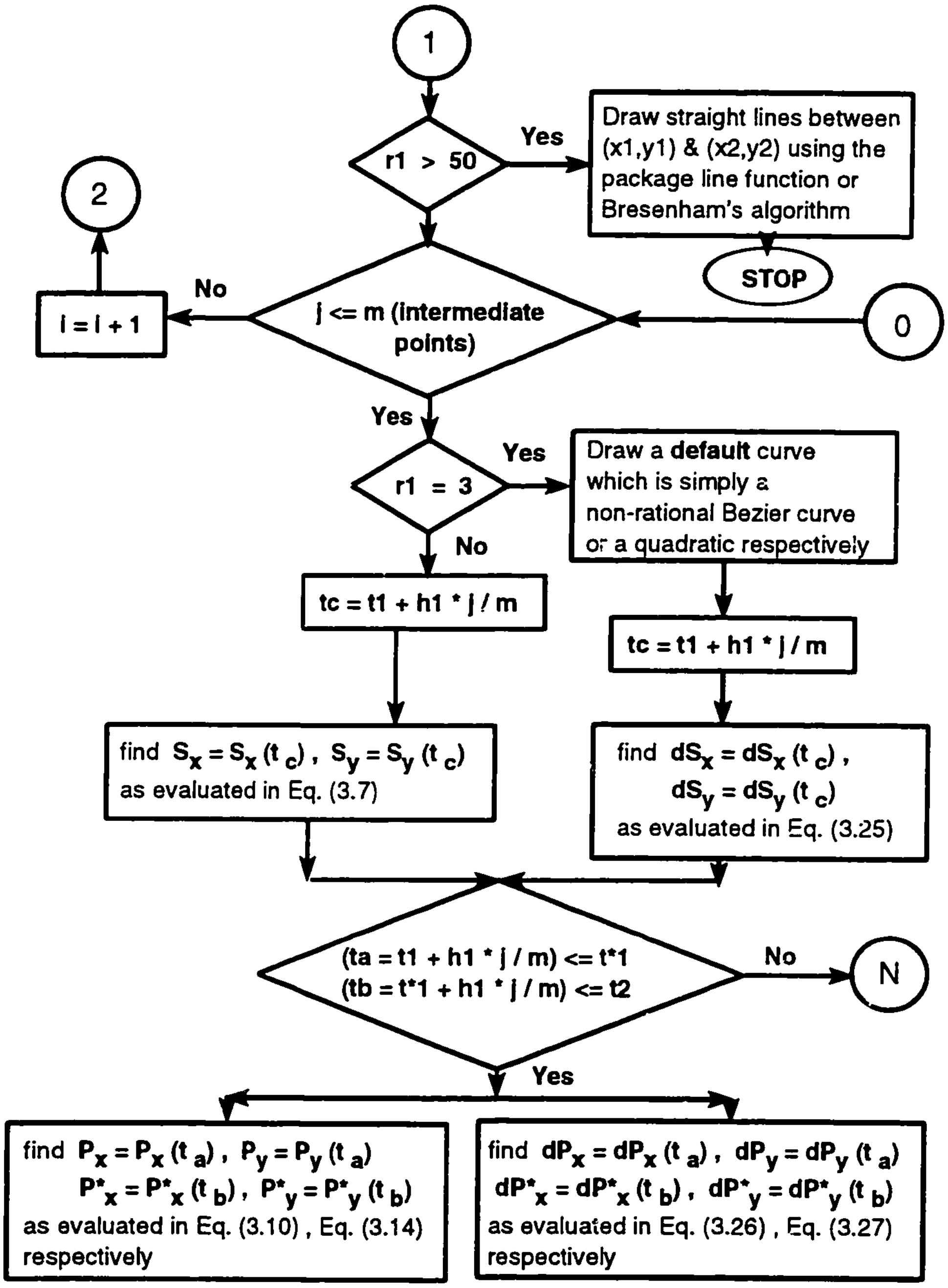
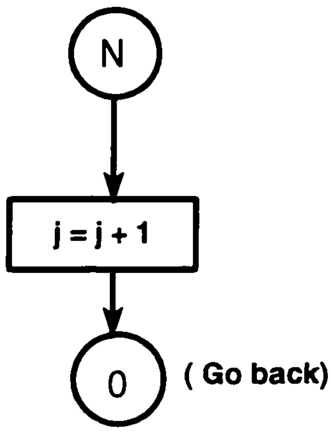


Figure 4.16: Flow-chart for obtaining a C^2 continuous rational cubic and conic. (*cont...*)



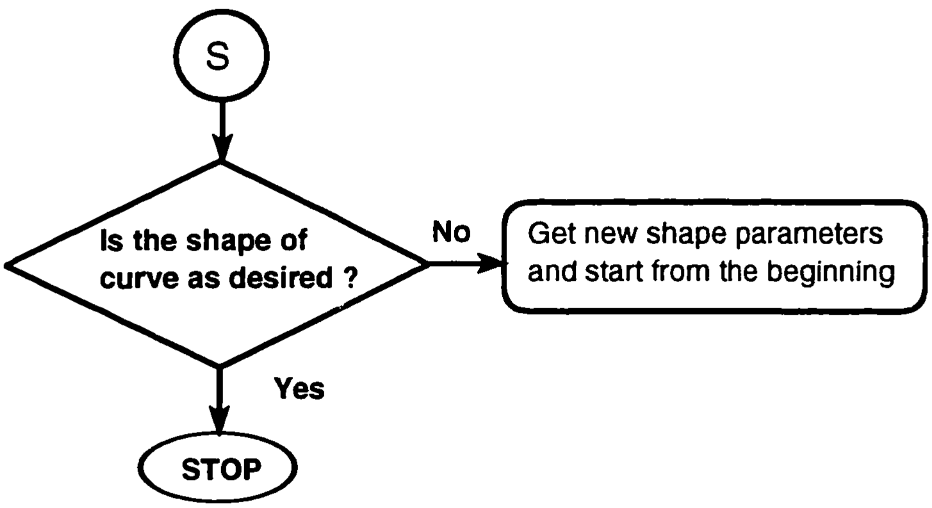


Figure 4.17: Flow-chart for obtaining a C^2 continuous rational cubic and conic. (*cont...*)

Chapter 5

Conversion of a rational cubic to an equivalent rational quadratic

This chapter deals with the conversion of a rational cubic into a conic. That is, unlike previous chapters, here, a rational cubic should first be drawn to get a conic. In this chapter another method of converting a rational cubic into a conic will be investigated. Previously, in [26] a Bezier curve was converted into a conic by using a least-squares method. In [26] each point of Bezier curve was substituted in the fundamental conic equation to get as many simultaneous equations which were then solved to get the conic control point and shape parameter. So, comparitively the method presented in this chapter is very fast than that of [26].

In the previous chapters the shape parameters for conic were given (*usually taken as half of rational cubic's shape parameter*) and the control points were found.

Also, the rational cubic was represented by two rational quadratics by splitting the rational cubic at half way through its parameter range.

- Here, both the shape parameter and the control points are found so that the conic approximates the rational cubic. Also, the rational cubic is represented by two rational quadratics by splitting the rational cubic not necessarily at the midpoint of its parameter range.

Before going to the actual conversion method, it is better to have a review of the implicit representationof a conic which is explained in chapter 2. For the sake of completeness the problem that was discussed in chapter 2 is again presented here. The problem is about **finding a conic from its control points plus another point that lies on the conic**. Let U_i^*, W_i, V_i^* be the three control points and $\mathbf{p}$ be a point on the rational quadratic. Now the implicit form of the rational quadratic is given by

$$\tau_1^2 = 4\gamma^2 \tau_0 \tau_2 \tag{5.1}$$

where the τ_i are barycentric coordinates of a point $\mathbf{p}$ on the rational quadratic with respect to the triangle U_i^*, W_i, V_i^* such that

$$\tau_0 = \frac{area(\mathbf{p}, W_i, V_i^*)}{area(U_i^*, W_i, V_i^*)}$$

$$\tau_1 = \frac{area(U_i^*, \mathbf{p}, V_i^*)}{area(U_i^*, W_i, V_i^*)}$$

$$\tau_2 = \frac{area(U_i^*, W_i, \mathbf{p})}{area(U_i^*, W_i, V_i^*)}$$

where

$$area(\mathbf{p}, W_i, V_i^*) = (p_x(W_{iy} - V_{iy}^*) - W_{ix}^*(p_y - V_{iy}^*) + V_{ix}^*(p_y - U_{iy}^*))$$

$$area(U_i^*, \mathbf{p}, V_i^*) = (U_{ix}^*(p_y - V_{iy}^*) - p_x(U_{iy}^* - V_{iy}^*) + V_{ix}^*(U_{iy}^* - p_y))$$

$$area(U_i^*, W_i, \mathbf{p}) = (U_{ix}^*(W_{iy} - p_y) - W_{ix}(U_{iy}^* - p_y) + p_x(U_{iy}^* - W_{iy}))$$

$$area(U_i^*, W_i, V_i^*) = (U_{ix}^*(W_{iy} - V_{iy}^*) - W_{ix}(U_{iy}^* - V_{iy}^*) + V_{ix}^*(U_{iy}^* - W_{iy}))$$

Thus, given a control polygon and an arbitrary point on the conic, we may find the shape control parameter γ by finding the barycentric coordinates of the point and then selecting the equation (5.1) for γ.

5.1 Concept of conversion

When deciding how to recover the rational cubic, two possibilities (as illustrated in *Figure 5.1* have to be considered.

- The first possibility occurs when the two endpoint tangents do not intersect. In this case, as depicted in *Figure 5.1(a)* the rational cubic needs to be subdivided into two halves, with two different conics representing each half. Altough in most cases (excluding rational cubics with inflection points) the rational cubic could be subdivided at midpoint of parameter range, there is no guarantee that splitting at this point will always result in conic sections that best-fit the original rational cubic. Thus, to effeciently recover rational cubic a measure of the quality of the conic-fit needs to be made at different splitting points along the rational cubic.

- The second possibility occurs, as shown in *Figure 5.1(b)* , when the two endpoint tangents do intersect. Although this rational cubic could be subdivided and approximated by two best-fitting conic sections, the need to do so will depend on whether a single conic approximation is within a desired best-fit criterion. That is, since an equivalent conic representation can be gained by chosing the intersection point (of the two endpoint tangents) as a control point and then using the x and y positions at mid-point of rational cubic to evaluate the required shape control parameter (by using the problem discussed earlier), the need to split such a rational cubic only becomes necessary if and when a single conic approximation is both numerically and aesthetically unacceptable.

The previous two possibilities does not encompass the rational cubic with inflection points (*points of zero curvature*). The best method to recapture a rational cubic with inflection points is to split the rational cubic and best place to split it is, at the inflection point itself. Thus, the point where the curvature of a rational cubic goes to zero, has to be located. For this the 2nd derivative of the ratioal cubic at each splitting point is used to test for the inflection point. As shown in *Figure 5.2*, this would result in the 2nd derivative being positive for a curve that is concave upwards, negative for a curve that is concave downwards and zero when the curve has zero curvature. The problem with this approach, however, is that it is only sufficient for rational cubics that do not have multiple data points within each segment; otherwise, as illustrated in *Figure 5.3(b)* the additional points could be taken as being points of inflection.

A much better and ideally suited method to attain curvature information is to use the following expression

$$k(\theta) = \frac{S_x'(\theta)S_y''(\theta) - S_y'(\theta)S_x''(\theta)}{(S_x'(\theta)^2 - S_y'(\theta)^2)^{3/2}} \tag{5.2}$$

where S_x', S_y' and S_x'', S_y'' are the first and second derivatives respectively of the rational cubic's x and y coordinate values at parameter t. The way the equation (5.2) is used to detect inflection points is by checking if the sign of the curvature value $k(\theta)$ changes within a rational cubic segment. If there is no change of sign then the curve does not contain an inflection point. If, on the other hand, a change of sign in $k(\theta)$ is detected then there exists an inflection point between the points where the sign change has taken place.

After finding the splitting point, which may be at the midpoint of rational cubic in case of no zero curvature or at any other point depending on the inflection point, the next step is to find the control points of the conic segments. The first control point of the rational cubic is the first control point of the first conic, the last control point of the rational cubic is the third control point of the second conic as shown in *Figure 5.2.* The last control point of the first conic segment and the first control point of the second conic segment, is the splitting point. The middle control point of the two conics need to be found. For this, the θ parameter value of the rational cubic where the splitting point is found, is to be checked. Let this parameter value be represented by ρ. Then,

- If the $\rho > 0.5$, the middle control point of the *first conic* is moved towards the second control point of the rational cubic along the straight line joining the first and second control points of rational cubic. Similarly, if the $\rho > 0.5$, the middle control point of the *second conic* is moved towards the fourth control point of the rational cubic along the straight line joining the third and fourth control points of rational cubic as shown in *Figure 5.2.*

- If the $\rho < 0.5$, the middle control point of the *first conic* is moved towards the first control point of the rational cubic along the straight line joining the first and second control points of rational cubic. Similarly, if the $\rho < 0.5$, the middle control point of the *second conic* is moved towards the third control point of the rational cubic along the straight line joining the third and fourth control points of rational cubic as shown in *Figure 5.2.*

Now that the control points for both conics (approximating the rational cubic) are found, any two points lying on the rational cubic have to be taken. *Here one point at parameter $\theta = 0.25$ and another point at parameter $\theta = 0.75$; are taken.*

Next the barycentric coordinates of the first point (*at* $\theta = 0.25$) with respect to the control polygon of the first conic, and the barycentric coordinates of the second point (*at* $\theta = 0.75$) with respect to the control polygon of the second conic; are found. Using the equation (5.1), the shape control parameters are found for both the conic segments. Once the shape control parameter and control values are obtained, the conic segments can be drawn.

If the conics does not best-fit the rational cubic, then the whole process of - moving the middle control points of the conics, finding the barycentric ccordinates of the two points on the rational cubic, finding the shape parameter; is repeated until tne two conics best-fit the rational cubic and no more improvement can be achieved.

The procedure can be described in the form of an algorithm. This algorithm is invariant with respect to translation, rotation and change of scale: Thus, if a given rational cubic is translated, rotated, or rescaled, the best-fitting conics to the transformed rational cubic are the same as those that would have resulted by appropriately translating, rotating and rescaling the best-fitting conics to the original cubic.

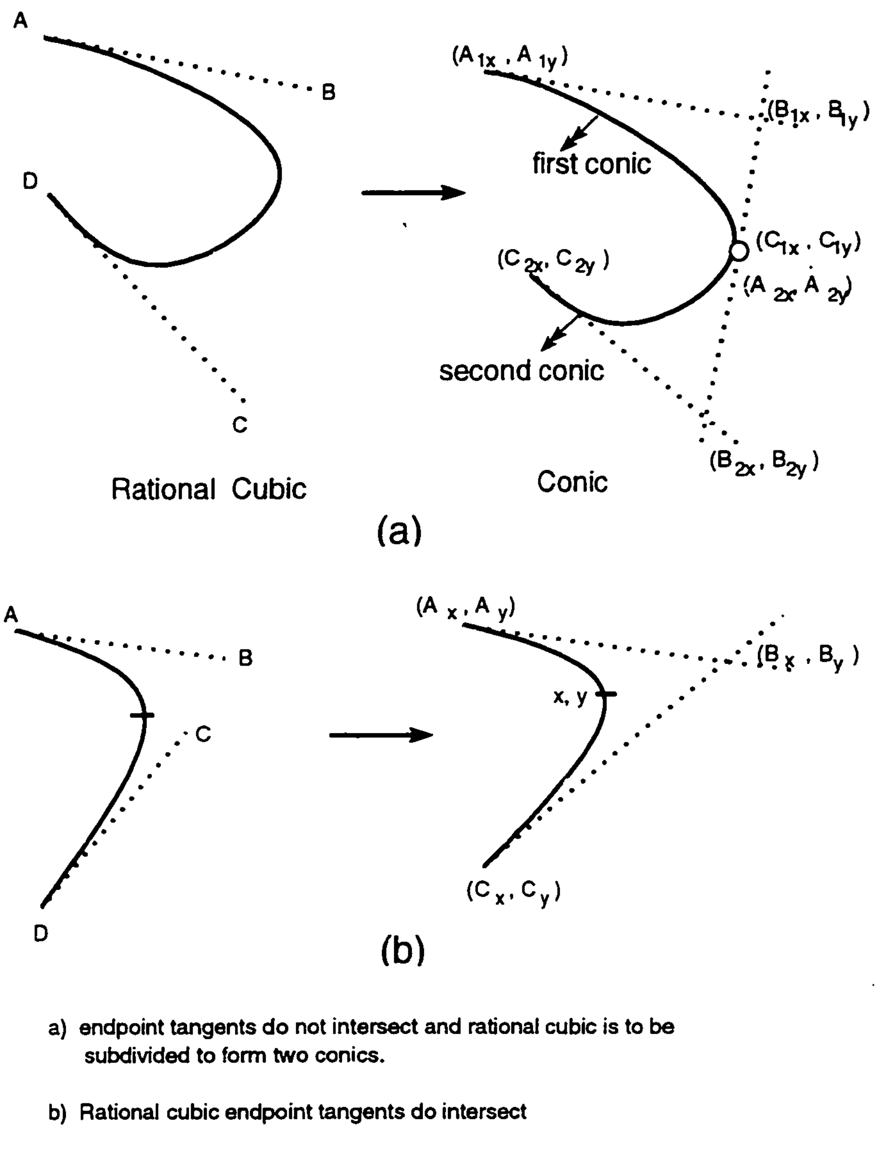


Figure 5.1: Two possibilities of conic rescue of rational cubic.

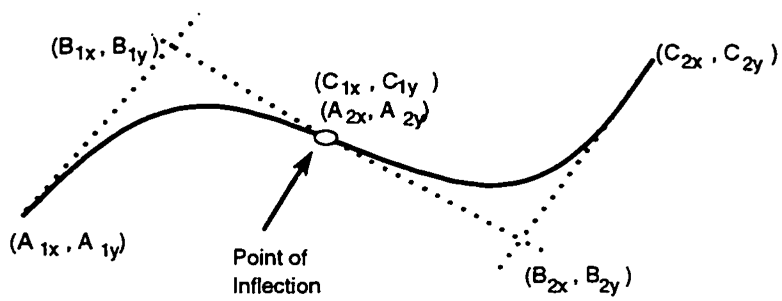
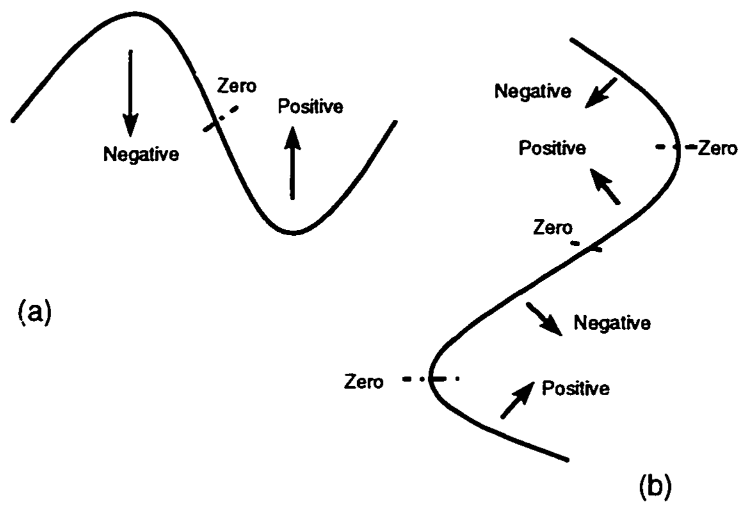


Figure 5.2: A rational cubic containing an inflection point is split at the inflection point.



a) a 'true' inflection point is returned when 2nd derivative becomes zero.

b) multiple point rational cubic where 2nd derivative returns three possible inflection points, though middle zero value is the actual point of inflection.

Figure 5.3: Sign of the 2nd derivative along a rational cubic containing an inflection point.

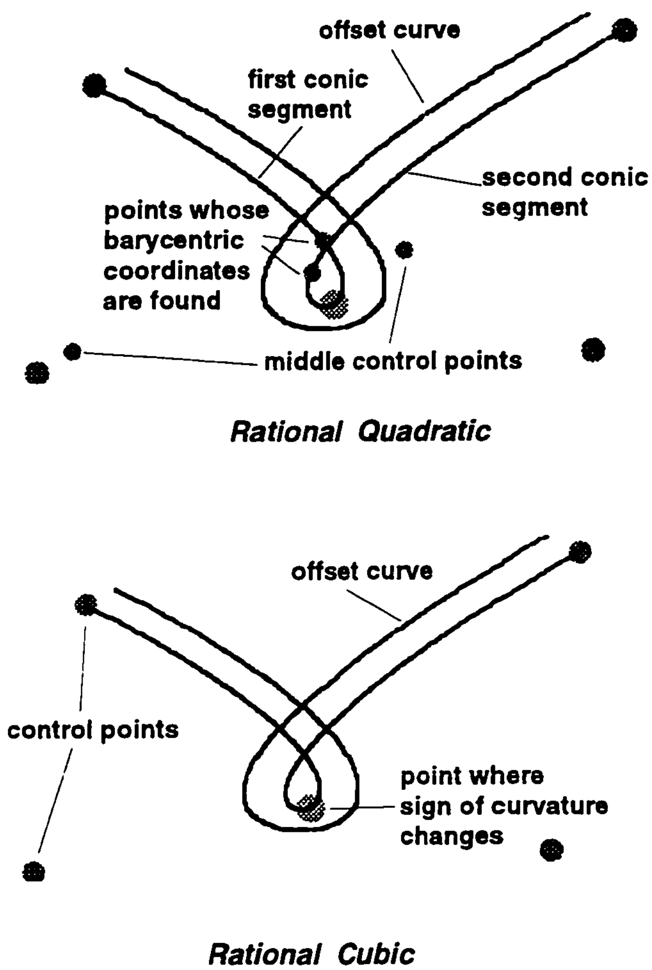


Figure 5.4: Rational cubic and its conic representation along with their offset curves.

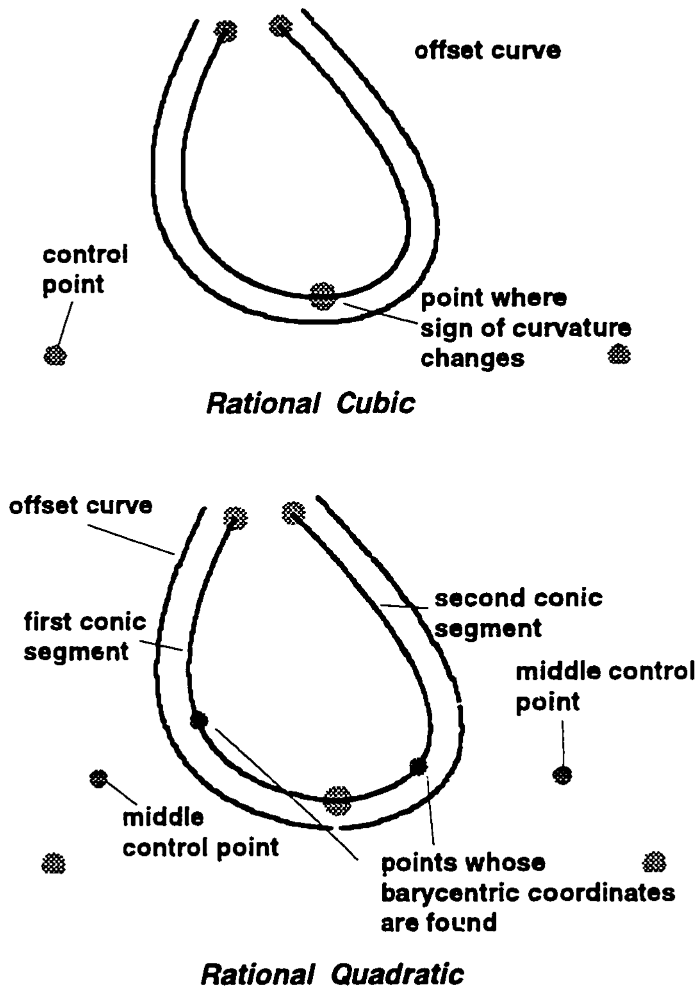


Figure 5.5: Rational cubic and its conic representation along with their offset curves.

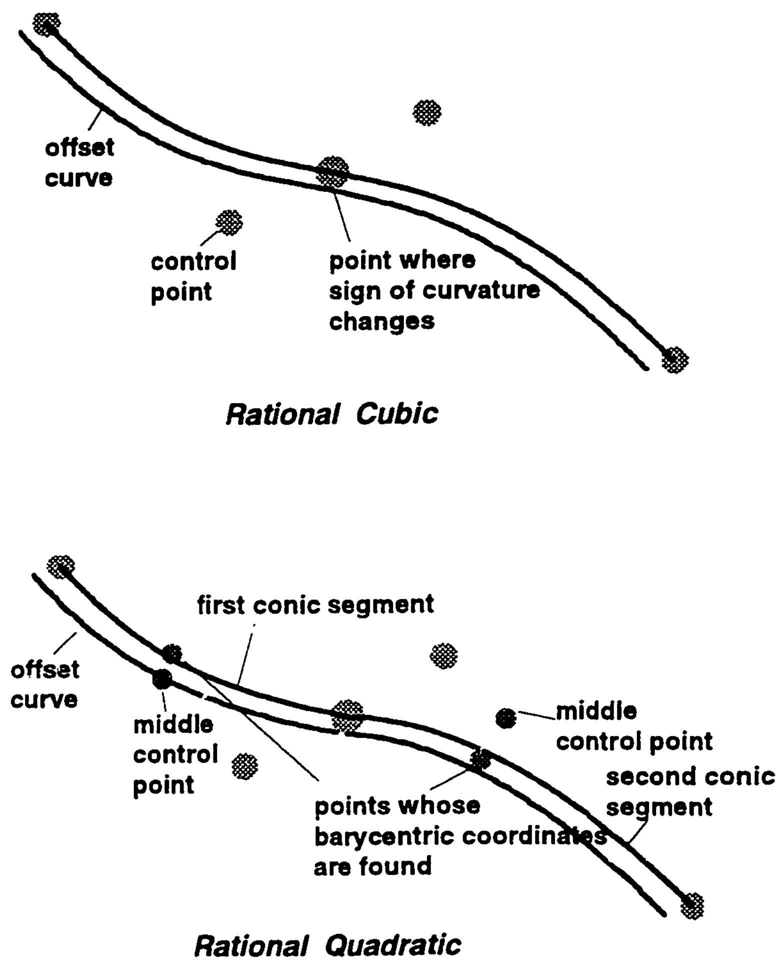


Figure 5.6: Rational cubic and its conic representation along with their offset curves.

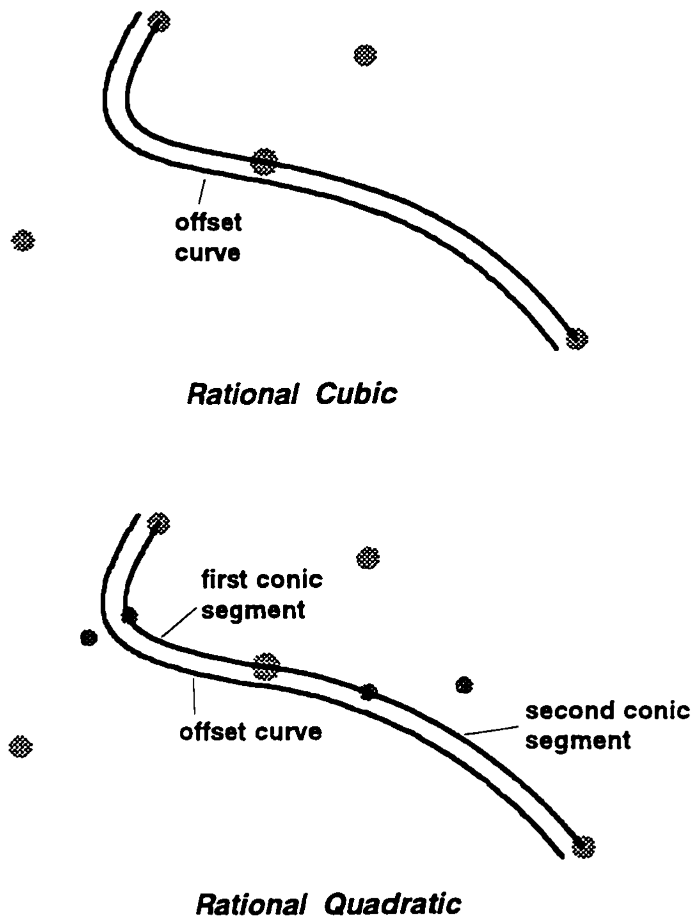


Figure 5.7: Rational cubic and its conic representation along with their offset curves.

Chapter 6

Surface Plotting

After the discussion of C^1 and C^2 continuous rational cubic and conic *curve* plotting, lets move on to the *surface* drawing of C^1 and C^2 continuous rational cubic and conic. The algorithm presented in this chapter is very simple and not at all complex as it is usually believed to be. The surface algorithm does not use any new methods than discussed in the previous chapters for curve drawing. It simply uses the curve drawing algorithm but in two phases with each phase plotting in two different directions.

In the first phase, the algorithm starts by reading an input file containing the following **3D** coordinate points (which are shown as a table, for simplicity). See *Table 6.1.*

Then, for each $Y_i, i = 1, 2, ..., n$, plot C^1 and C^2 continuous rational cubic and

	X_1	X_2	X_3	....	X_m
Y_1	Z_{11}	Z_{12}	Z_{13}	....	Z_{1m}
Y_2	Z_{21}	Z_{22}	Z_{23}	....	Z_{2m}
⋮	⋮	⋮	⋮	⋮	⋮
Y_n	Z_{n1}	Z_{n2}	Z_{n3}	....	Z_{nm}

Table 6.1: Input File containing the 3D coordinate points.

conic curves by taking the data points

$$(X_1, Z_{i1}), \quad (X_2, Z_{i2}), \quad ,....,(X_m, Z_{im})$$

as the user given control points $F_i, F_{i+1}, ..., F_{n+1}$ through which the rational cubic and conic passes. Use these data points in the equations of C^1 or C^2 continuous rational cubic and conic and continue with the method of drawing curves using the algorithms presented in the previous chapters for curves. When the curves are drawn in the **XZ** direction keeping **Y** direction constant, the intermediate points will be generated *at each parameter value of* t as:

The intermediate points between (X_1, Z_{i1}) and (X_2, Z_{i2}) can be represented as:

$$(X_{11}, Z_{11,1}), \quad (X_{12}, Z_{11,2}), \quad ,....,(X_{1u}, Z_{11,u})$$

and likewise for intermediate points between other data points. the u is the number

Y_1	$X_1 = X_{11}, X_{12}, ..., X_{1u}$	....	$X_m = X_{m1}, X_{m2}, ..., X_{mu}$
	$Z_{11} = Z_{11,1}, ..., Z_{11,u}$	....	$Z_{1m} = Z_{1m,1}, ..., Z_{1m,u}$
. . .	. . .	. . .	. . .
Y_n	$Z_{n1} = Z_{n1,1}, ..., Z_{n1,u}$	....	$Z_{nm} = Z_{nm,1}, ..., Z_{nm,u}$

Table 6.2: Output File containing the 3D cocrdinate points.

of intermediate points generated between any two data points. This u should be fixed for intermediate points between all data points. For example, if 5 intermediate points are taken between any two data points then the parameter t should start at t_i and end at t_i with in between increment being $\frac{t_{i+1}-t_i}{5}$.

In a general form, the intermediate points between $(X_j, Z_{ij}), (X_{j+1}, Z_{ij+1})$ corresponding to Y_i, will be

$$(X_{i1}, Z_{ij,1}), \quad (X_{i2}, Z_{ij,2}), \quad ....., (X_{iu}, Z_{ij,u})$$

where $i = 1, 2, ..., n-1$ and $j = 1, 2, ..., m-1$. Thus, here a total of $n \times u \times m$ points will be generated. After the required intermediate points are generated between all data points corresponding to each Y_i, the algorithm will generate file (See *Table 6.2*) as follows : The points generated, as given in the table, are then given to a package to draw curves. The shape parameters r_i and γ_i can be different between different segments for a particular $Y_i, i = 1, 2, ..., n$ but the same set of shape parameters should be used for segments of any other Y_i. Thus, in the first phase curves are drawn in one direction i.e., in **XZ** direction and a new set of points are written to a file which will then be used by the next phase.

In the second phase, the algorithm reads the output file generated in the previous phase, and plots curves in the **YZ** direction by taking the intermediate points u as explained in the first phase. The **X** direction is kept constant for each set of points.

The algorithm plot curves on the data points $(Y_1, Z_{1i,u}), (Y_2, Z_{2i,u}), ..., (Y_n, Z_{ni,u})$ for each X_{iu}, where $i = 1, 2, ..., n$ and u is the number of intermediate points generated in each interval of the parameter t. These data points are taken as the user given control points $F_i, F_{i+1}, ..., F_{n+1}$ from which the curve passes. Use these data points in the equations of C^1 or C^2 continuous rational cubic and conic and continue with the method of drawing curves using the algorithms presented in the previous chapters for curves. Here, a total of $n \times m \times u \times (m-1)$ points will be generated. The intermediate points thus generated in the **YZ** direction will be written to a file and given to the package to plot the curves. Here also the shape parameters $r_i$and γ_i can be different between segments for a particular $X_{iu}, i = 1, 2, ..., m-1$ but the same set of parameters should be used for segments of any other X_{iu}.

Thus, the curves plotted in the two phases in two different directions keeping the other direction constant, results in the surface plot of any object using any of

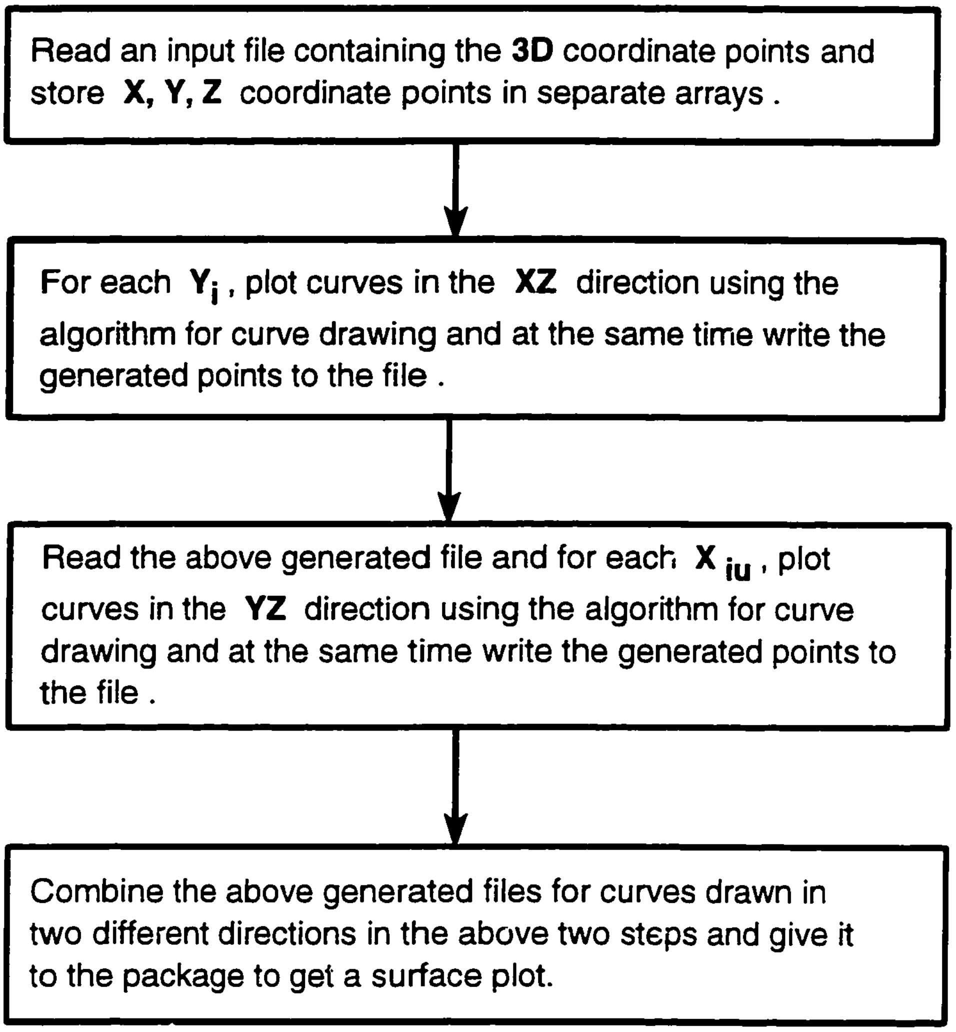


Figure 6.1: Flow-chart explaining the method of obtaining a surface plot.

the C^1 or C^2 continuous rational cubic or conic.

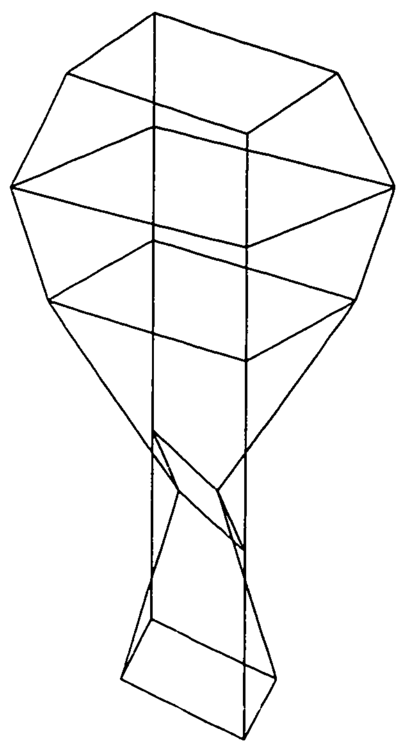


Figure 6.2: Control Polygon of the *cup* shaped object shown in the following demonstrations

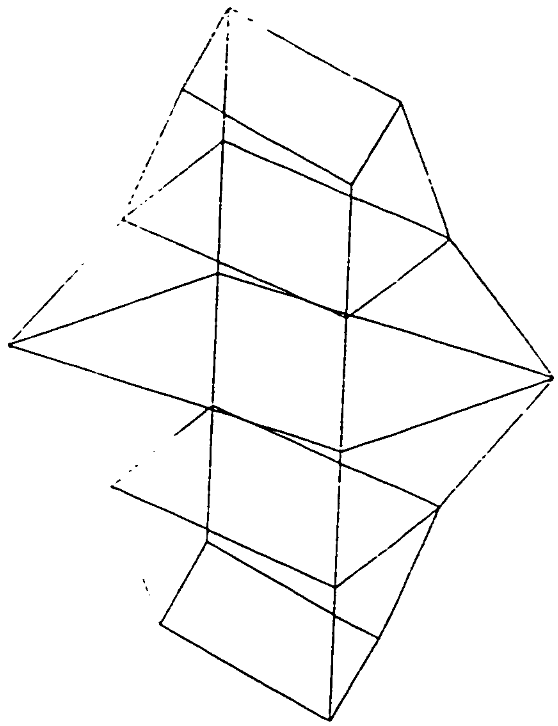


Figure 6.3: Control Polygon of the *vase* shaped object shown in the following demonstrations

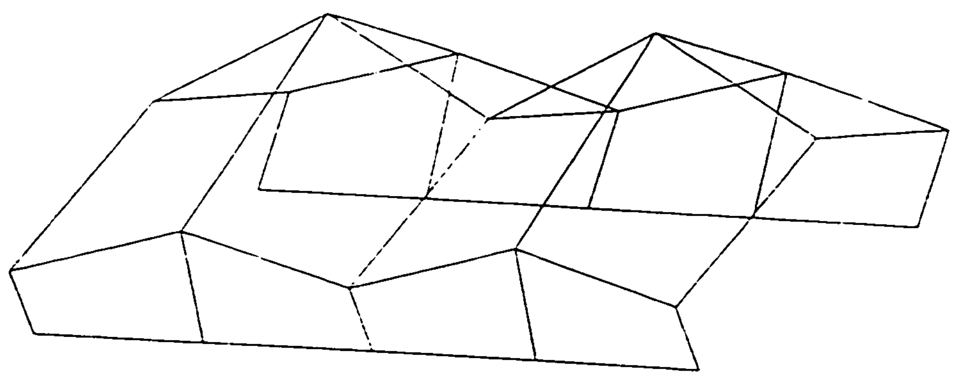


Figure 6.4: Control Polygon of the *hat* shaped object shown in the following demonstrations

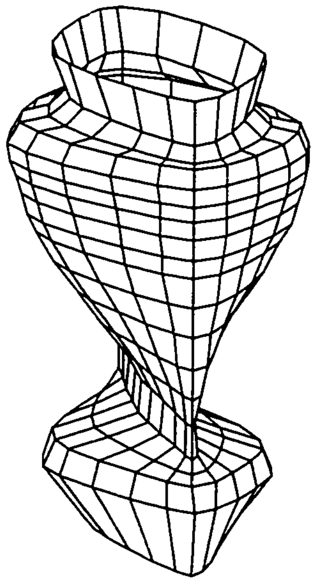


Figure 6.5: Object drawn by using C^1 continuous rational cubic with (*Default shape parameters*)

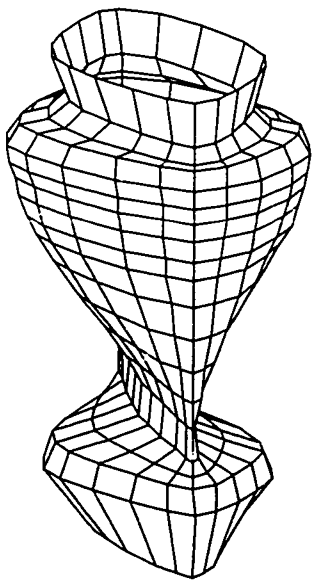


Figure 6.6: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with (*Default shape parameters*)

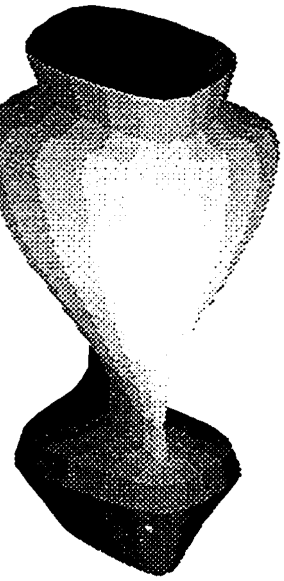


Figure 6.7: Object drawn by using C^1 continuous rational cubic with *Default shape parameters*

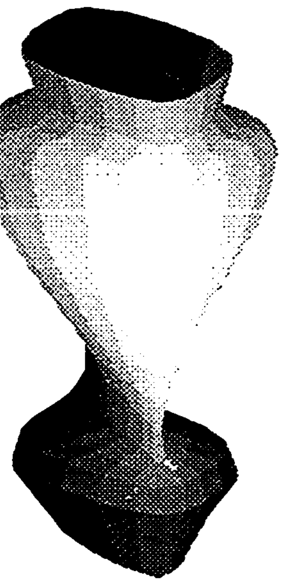


Figure 6.8: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Default shape parameters*

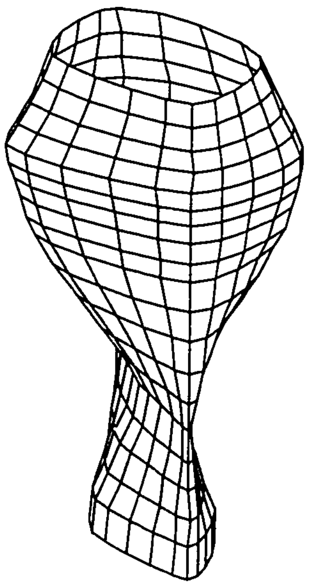


Figure 6.9: Object drawn by using C^1 continuous rational cubic with *Interval tension applied to the shape parameters for the top and bottom segments*

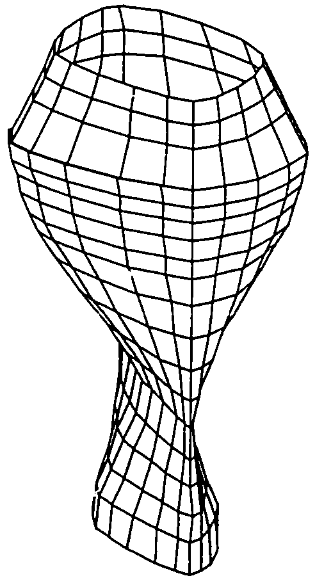


Figure 6.10: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Interval tension applied to the shape parameters for the top and bottom segments*

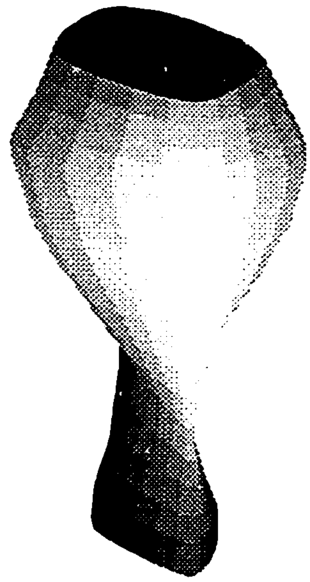


Figure 6.11: Object drawn by using C^1 continuous rational cubic with *Interval tension applied to the shape parameters for the top and bottom segments*

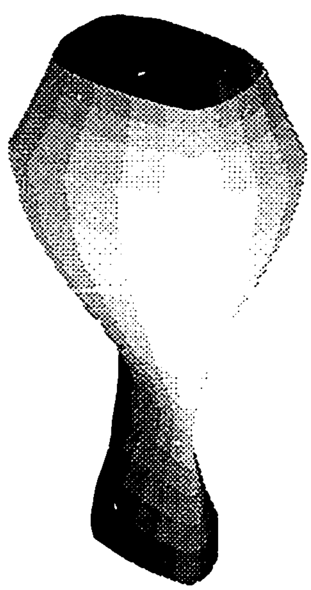


Figure 6.12: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Interval tension applied to the shape parameters for the top and bottom segments*

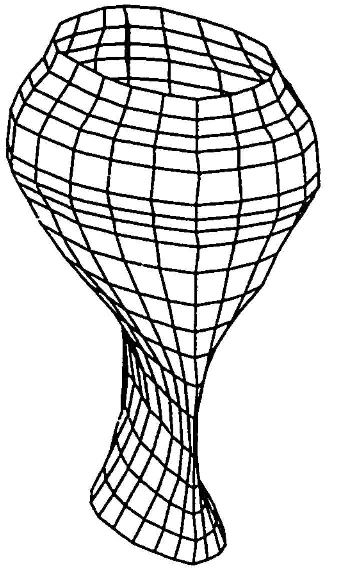


Figure 6.13: Object drawn by using C^2 continuous rational cubic with *Default shape parameters*

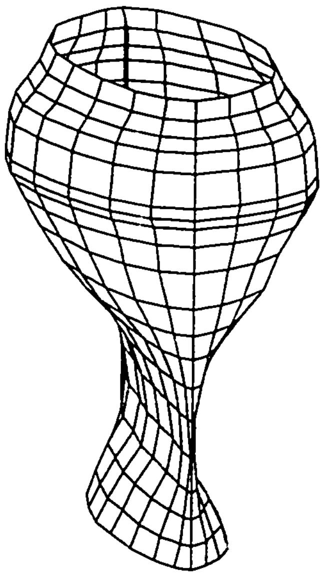


Figure 6.14: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Default the shape parameters*

Figure 6.15: Object drawn by using C^2 continuous rational cubic with *Default shape parameters*

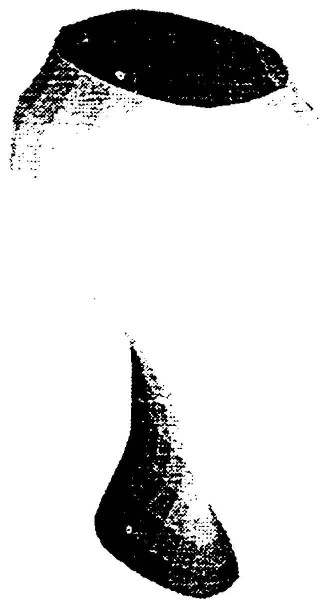


Figure 6.16: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with with *Default shape parameters*

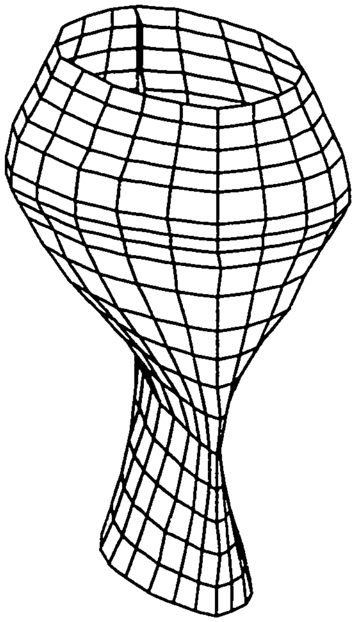


Figure 6.17: Object drawn by using C^2 continuous rational cubic with *Interval tension applied to the shape parameters for the top and bottom segments*

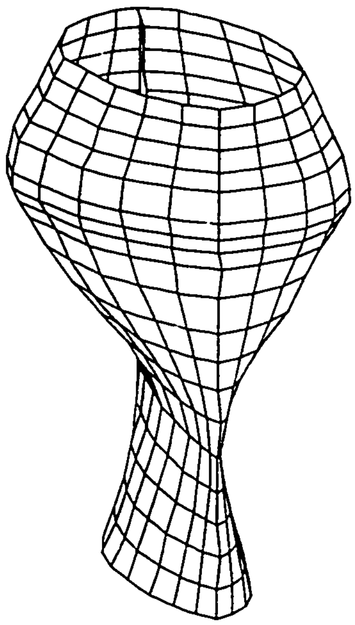


Figure 6.18: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Interval tension applied to the shape parameters for the top and bottom segments*

Figure 6.19: Object drawn by using C^2 continuous rational cubic with *Interval tension applied to the shape parameters for the top and bottom segments*

Figure 6.20: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Interval tension applied to the shape parameters for the top and bottom segments*

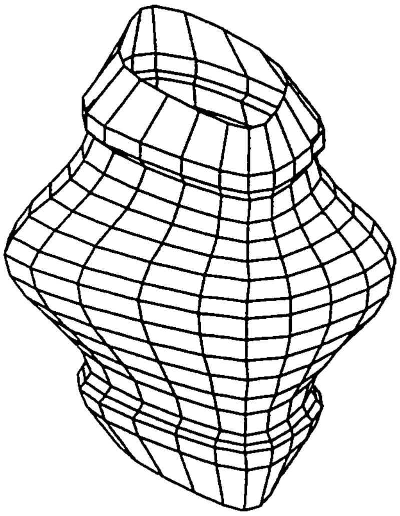


Figure 6.21: Object drawn by using C^1 continuous rational cubic with *Default shape parameters*

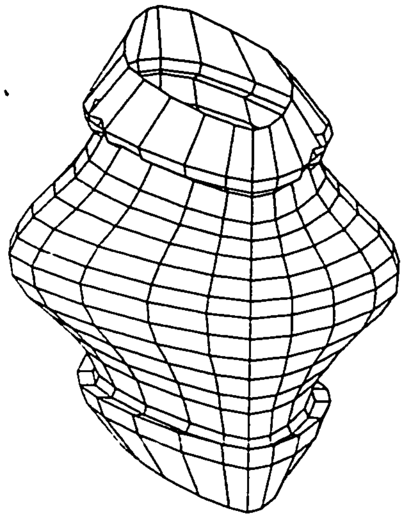


Figure 6.22: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Default shape parameters*

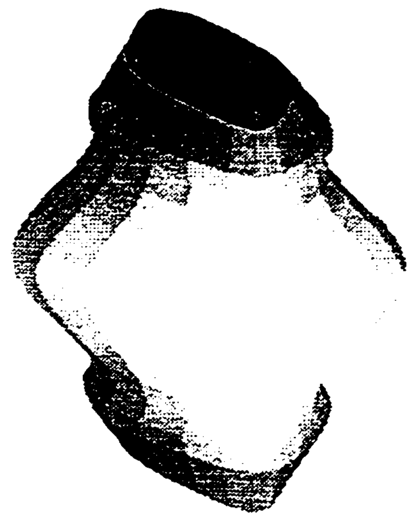


Figure 6.23: Object drawn by using C^1 continuous rational cubic with *Default shape parameters*

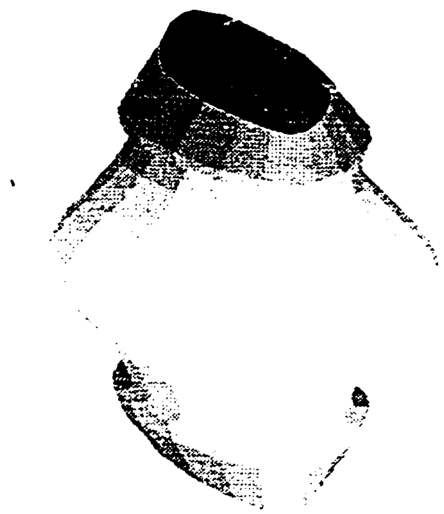


Figure 6.24: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Default shape parameters*

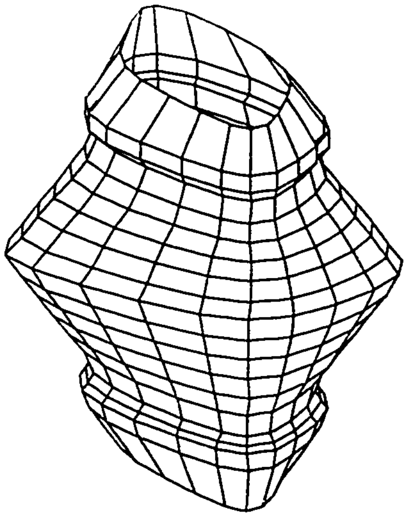


Figure 6.25: Object drawn by using C^1 continuous rational cubic with *Point tension applied to the shape parameters of the middle two adjacent segments*

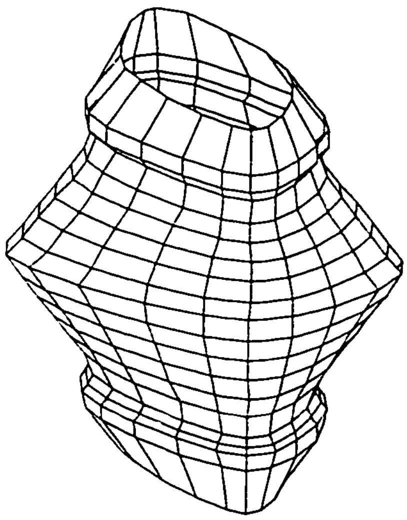


Figure 6.26: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Point tension applied to the shape parameters of the middle two adjacent segments*

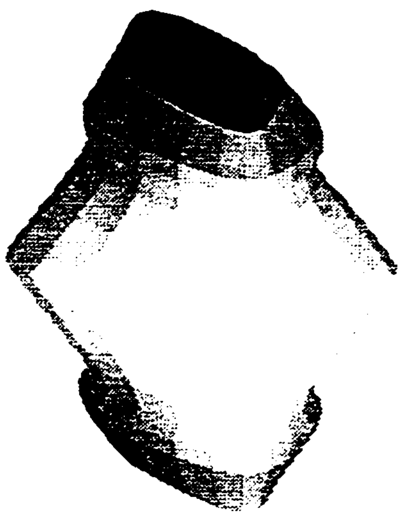


Figure 6.27: Object drawn by using C^1 continuous rational cubic with *Point tension applied to the shape parameters of the middle two adjacent segments*

Figure 6.28: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Point tension applied to the shape parameters of the middle two adjacent segments*

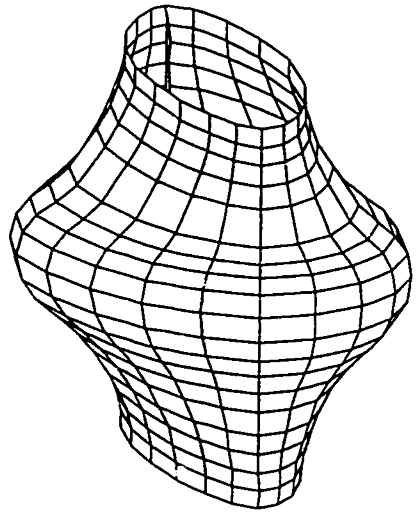


Figure 6.29: Object drawn by using C^2 continuous rational cubic with *Default the shape parameters*

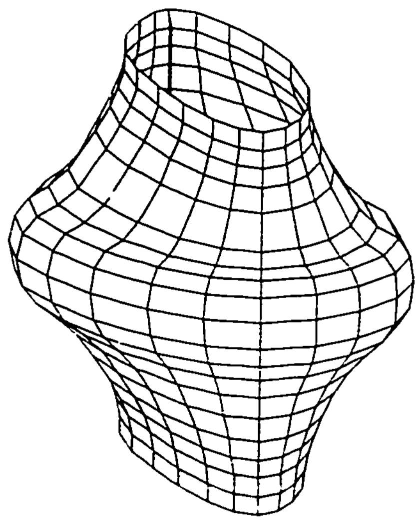


Figure 6.30: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Default shape parameters*

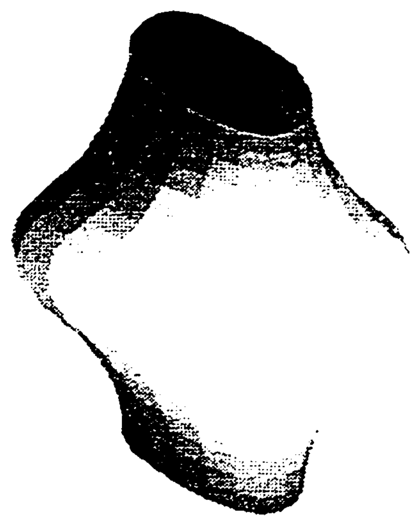


Figure 6.31: Object drawn by using C^2 continuous rational cubic with *Default the shape parameters*

Figure 6.32: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Default shape parameters*

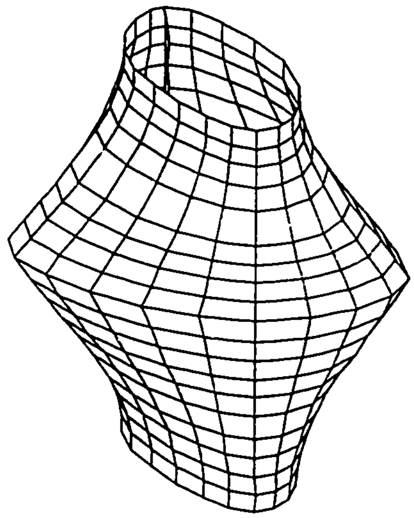


Figure 6.33: Object drawn by using C^2 continuous rational cubic with *Point tension applied to the shape parameters of the middle two adjacent segments*

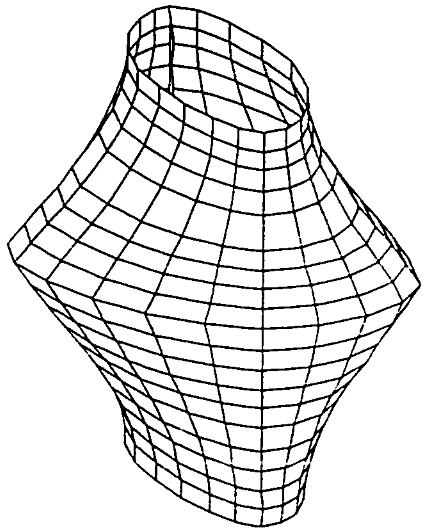


Figure 6.34: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Point tension applied to the shape parameters of the middle two adjacent segments*

Figure 6.35: Object drawn by using C^2 continuous rational cubic with *Point tension applied to the shape parameters of the middle two adjacent segments*

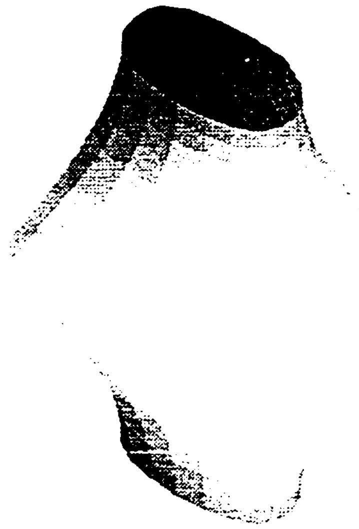


Figure 6.36: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Point tension applied to the shape parameters of the middle two adjacent segments*

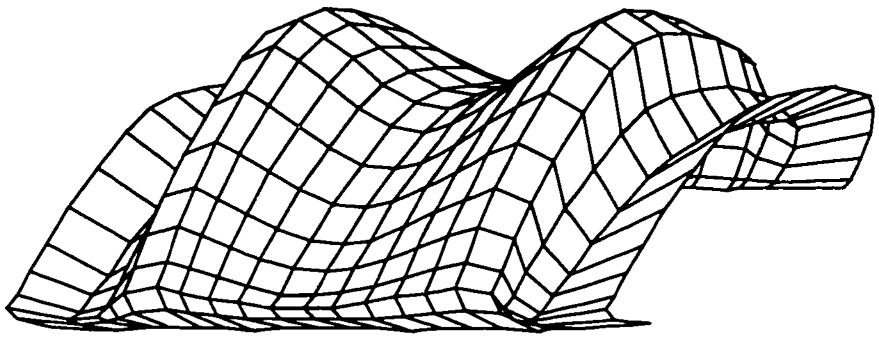


Figure 6.37: Object drawn by using C^1 continuous rational cubic with *Default shape parameters*

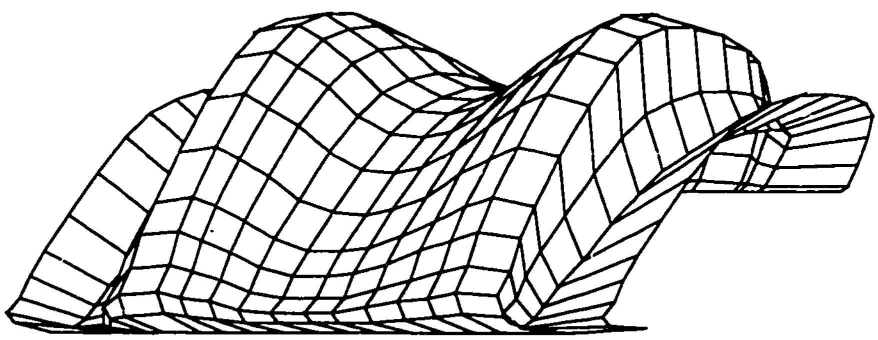


Figure 6.38: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Default shape parameters*

Figure 6.39: Object drawn by using C^1 continuous rational cubic with *Default shape parameters*

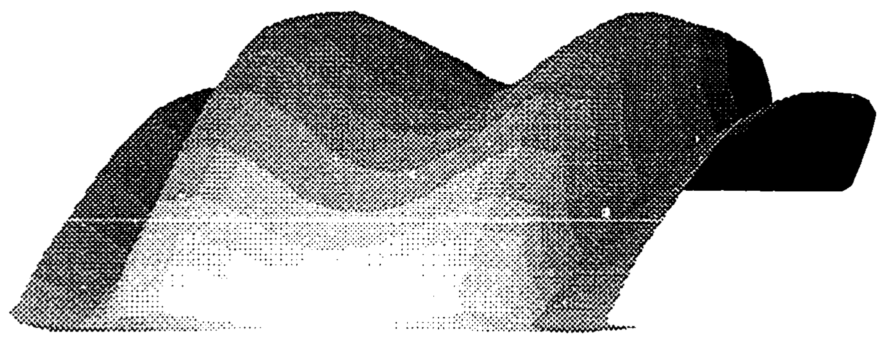


Figure 6.40: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Default shape parameters*

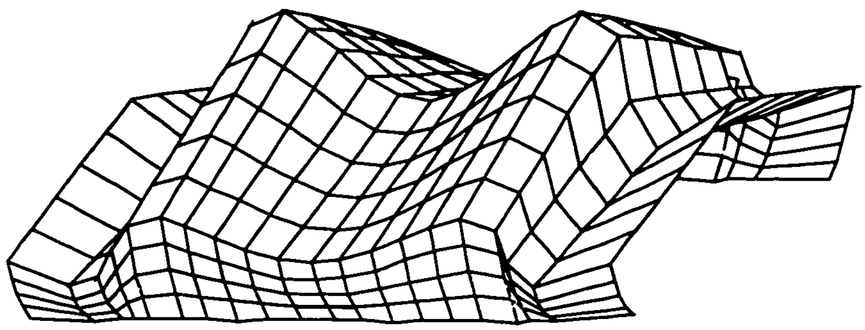


Figure 6.41: Object drawn by using C^1 continuous rational cubic with *Global tension in one direction*

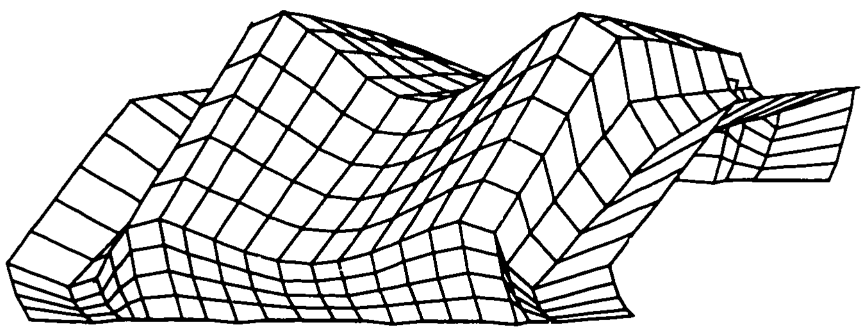


Figure 6.42: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Global tension in one direction*

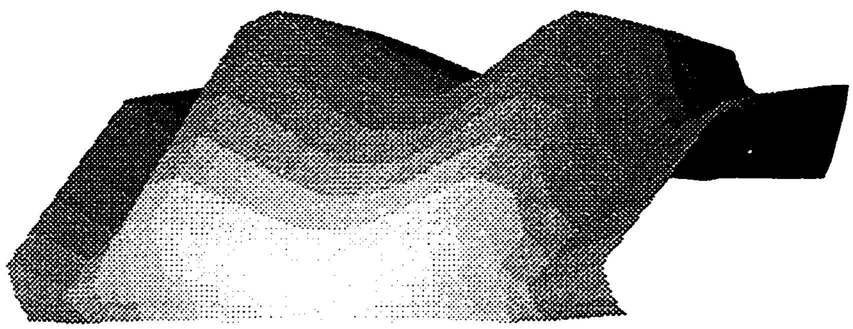


Figure 6.43: Object drawn by using C^1 continuous rational cubic with *Global tension in one direction*

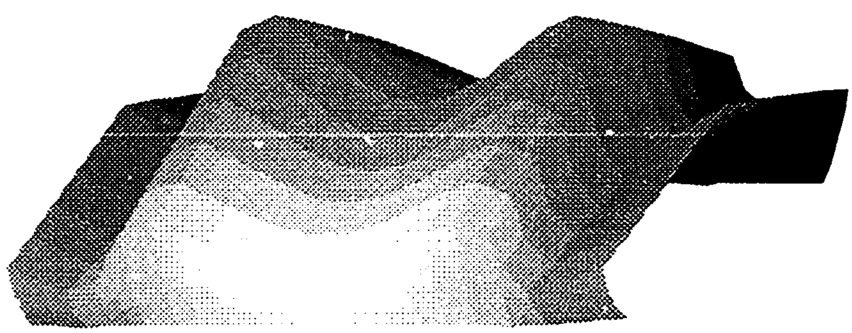


Figure 6.44: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Global tension in one direction*

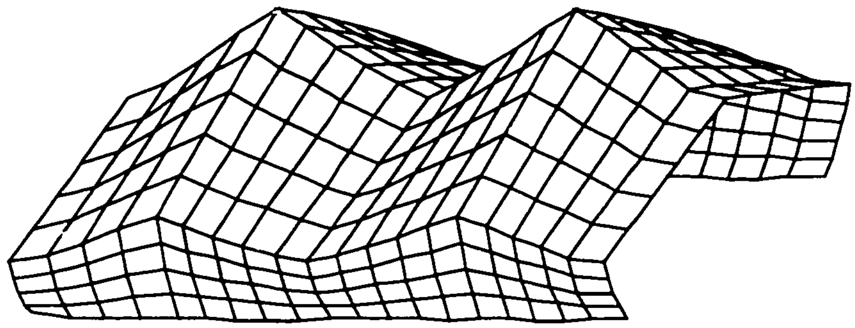


Figure 6.45: Object drawn by using C^1 continuous rational cubic with *Global tension in both directions*

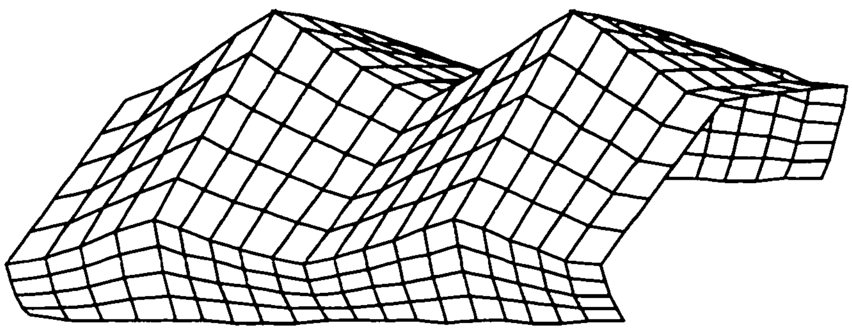


Figure 6.46: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Global tension in both directions*

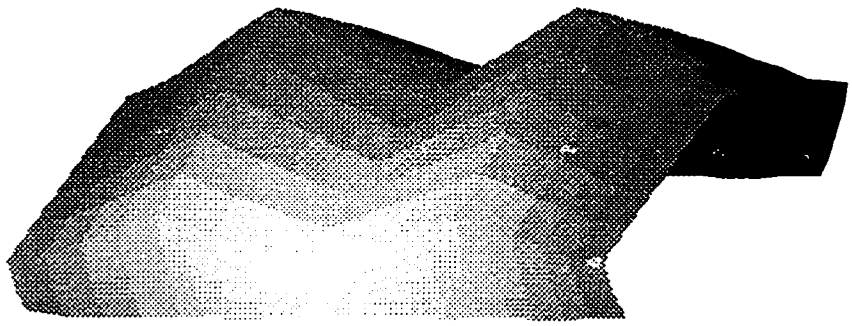


Figure 6.47: Object drawn by using C^1 continuous rational cubic with *Global tension in both directions*

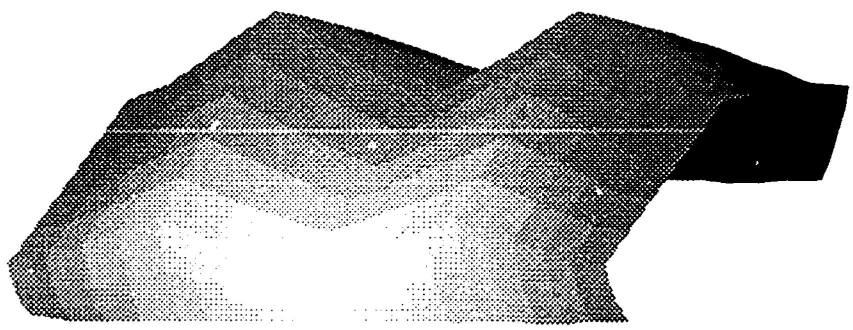


Figure 6.48: Object drawn by using C^1 continuous conic (similar to the above rational cubic) with *Global tension in both directions*

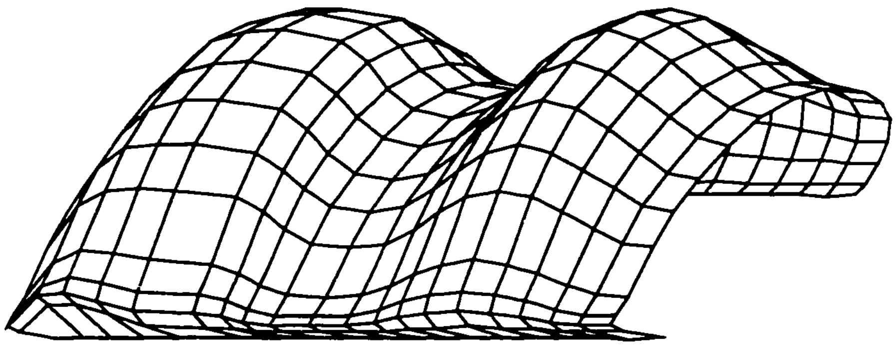


Figure 6.49: Object drawn by using C^2 continuous rational cubic with *Default shape parameters*

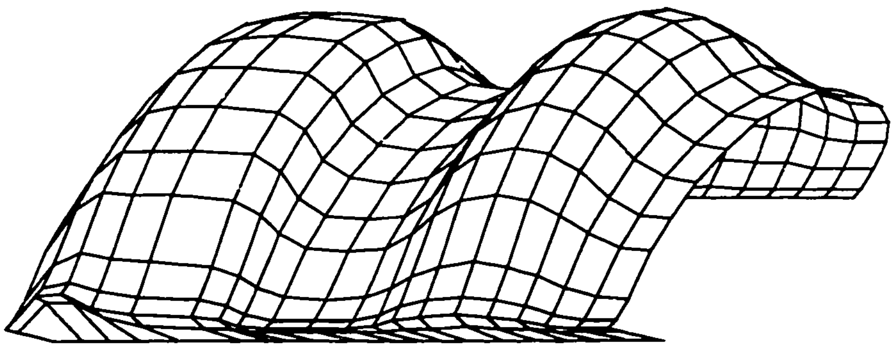


Figure 6.50: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Default shape parameters*

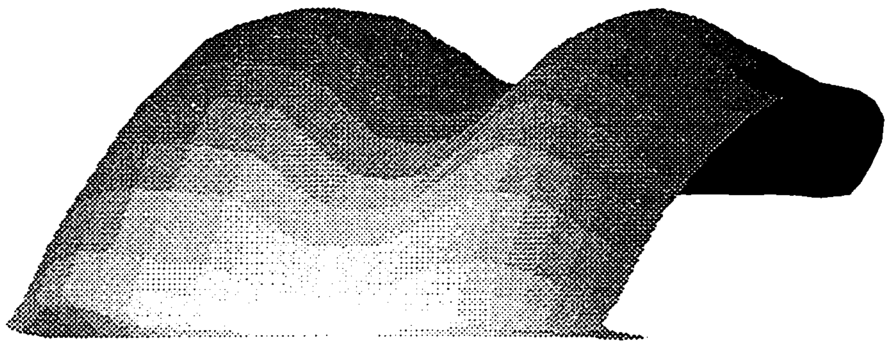


Figure 6.51: Object drawn by using C^2 continuous rational cubic with *Default shape parameters*

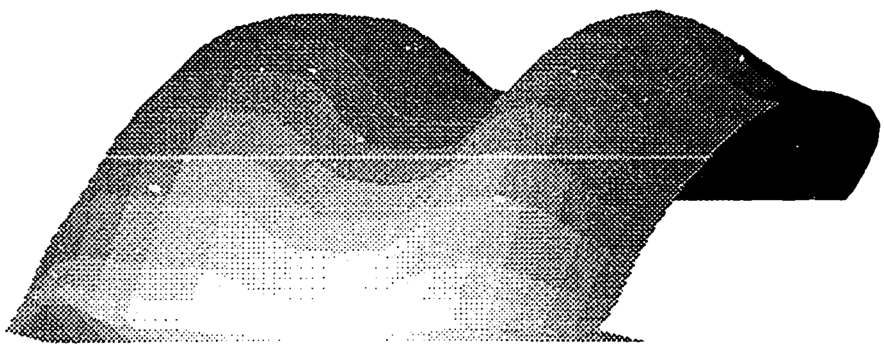


Figure 6.52: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Default shape parameters*

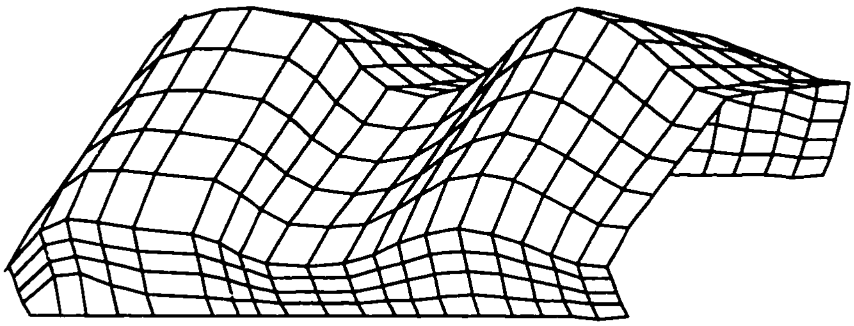


Figure 6.53: Object drawn by using C^2 continuous rational cubic with *Global tension in one direction*

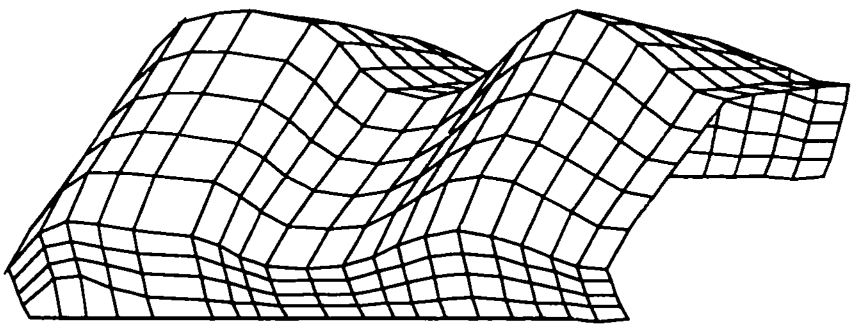


Figure 6.54: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Global tension in one direction*

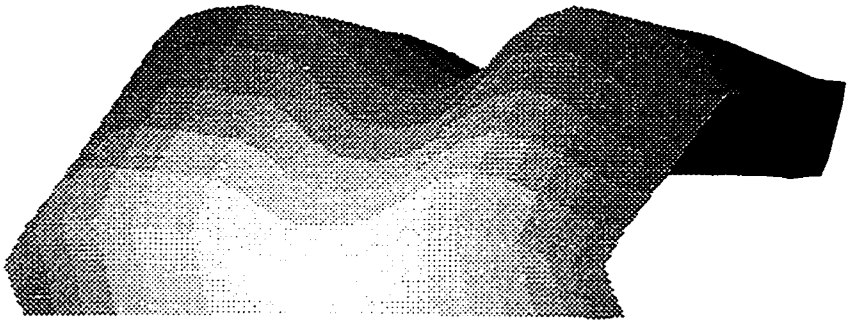


Figure 6.55: Object drawn by using C^2 continuous rational cubic with *Global tension in one direction*

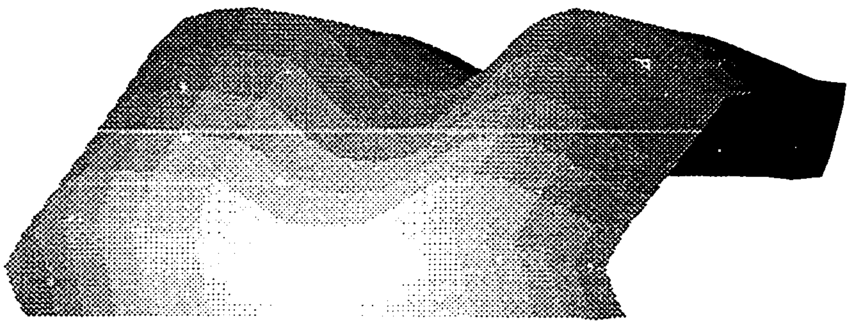


Figure 6.56: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Global tension in one direction*

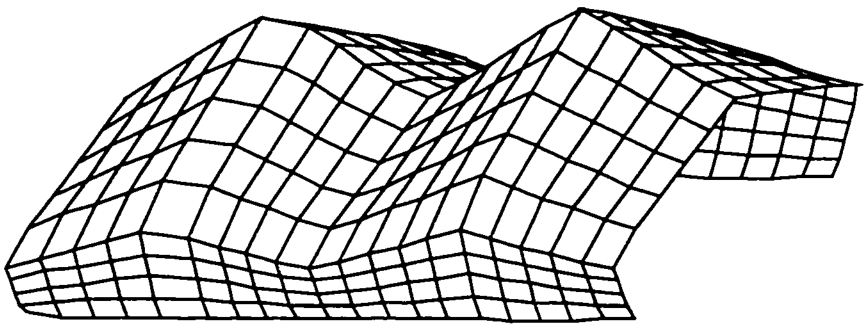


Figure 6.57: Object drawn by using C^2 continuous rational cubic with *Global tension in both directions*

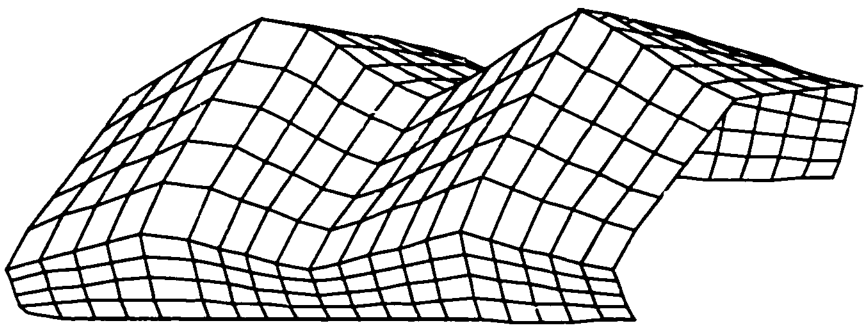


Figure 6.58: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Global tension in both directions*

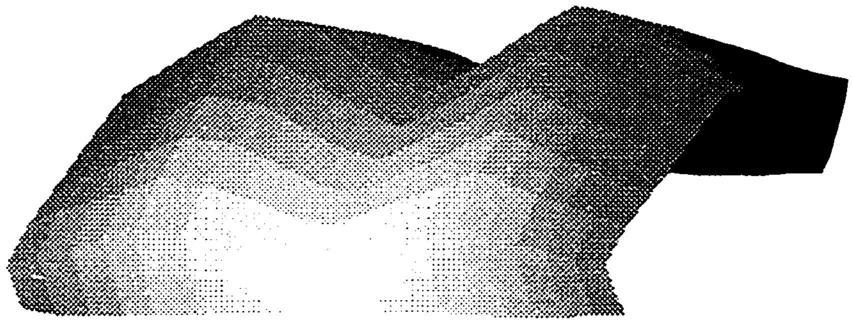


Figure 6.59: Object drawn by using C^2 continuous rational cubic with *Global tension in both directions*

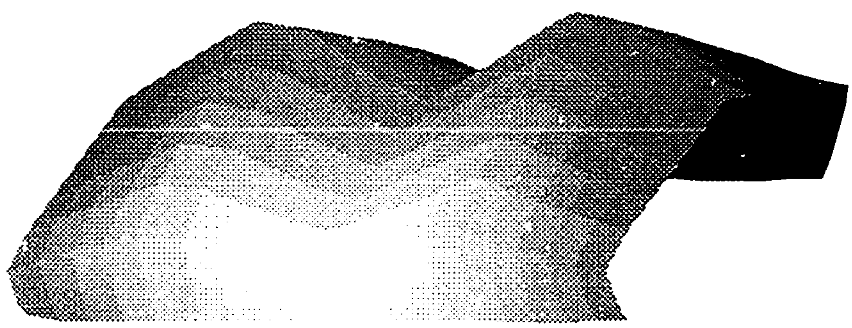


Figure 6.60: Object drawn by using C^2 continuous conic (similar to the above rational cubic) with *Global tension in both directions*

Chapter 7

CONCLUSIONS AND FUTURE WORK

7.1 CONCLUSIONS

As discussed in the problem statement, the main objective of proving that any curve or object drawn by a rational cubic can well be equivalently drawn by a conic, is attained. This objective has not only been attained for C^1 continuous curves but for C^2 continuous curves as well, as the figures of the previous chapters demonstrate. It should be noted that though the representation of the conic is discussed along with that of rational cubic, it does not imply that one should first construct a rational cubic and then only can represent a conic. The conic is discussed along with rational cubic just to prove how a conic can well be used to represent any shape which till now has been represented by rational cubic.

Also, it has been proved how the properties of rational cubic can well be applied to the conic to achieve the same effect. The different cases considered like - the open curve, close curve, relaxed end, normalized end, clamped end etc..., along with the different parameters affecting the shape of the curve goes way beyond to prove that whatever might be the case one can still get an equivalent conic representation for a rational cubic.

The method of computation of both rational cubic and conic is very simple and computationally fast compared to the other methods, as in [26]. This fast method of generating curves and surfaces is very much the need of the industry where very complex and huge drawings are the need of the hour. Consider, the designing of an airplane, a car, a ship, or for that matter the brain of a human body. All of these require an accurate and complex drawings.

The study of the different properties of conic and the different ways of drawing a conic also provide much flexibility in representing curves and surfaces where given any input conditions one can still obtain a conic of the desired shape.

Also discussed are the different ways of obtaining **offsets** to conics. The applications of drawing offsets are many and the most important of them is *Font designing*. Nowadays, with so many word-processors around one cannot imagine a one without beautiful Fonts incorporated in them. Such is the importance of Fonts in the day-to-day life.

After proving how to get an equivalent conic representation of a rational cubic let's summarize the advantages of conics over rational cubic :

- It is much **easier** to find the intersection of a line with a conic than with a rational cubic. The finding of intersection of a line with a curve is very much used in the determination of hidden surface. This is one of the most important advantages of a conic considering that almost all the objects are finally viewed by removing their hidden surfaces and imagine how difficult it will be for a very complex and detailed object.

- The shape control is **twice stronger** in conic than in rational cubic. This is very important when the desired shape is not obtained and the object has to be redrawn with a new shape parameter. That is, the number of redrawings done with conic is less compared to that of rational cubic. Also it will take less computation time because of smaller value.

- Above all, the computational requirements are **less** with conic than with rational cubic, because conic has one degree less than rational cubic.

7.2 FUTURE WORK

- Geometric continuity can be incorporated in addition to the parametric continuity.

- Biased shape parameters can be incorporated, apart from point and interval tension effects.

- The work done in [26] can be included to check if the accuracy of conic representation compared to that of rational cubic can be achieved as outlined in *Figures 7.1 and 7.2.* That is, the initial checking for inflection points and splitting the rational cubic at the inflection point as done in [26] can be incorporated. In the work discussed in the previous chapters the rational cubic is split at its mid-point.

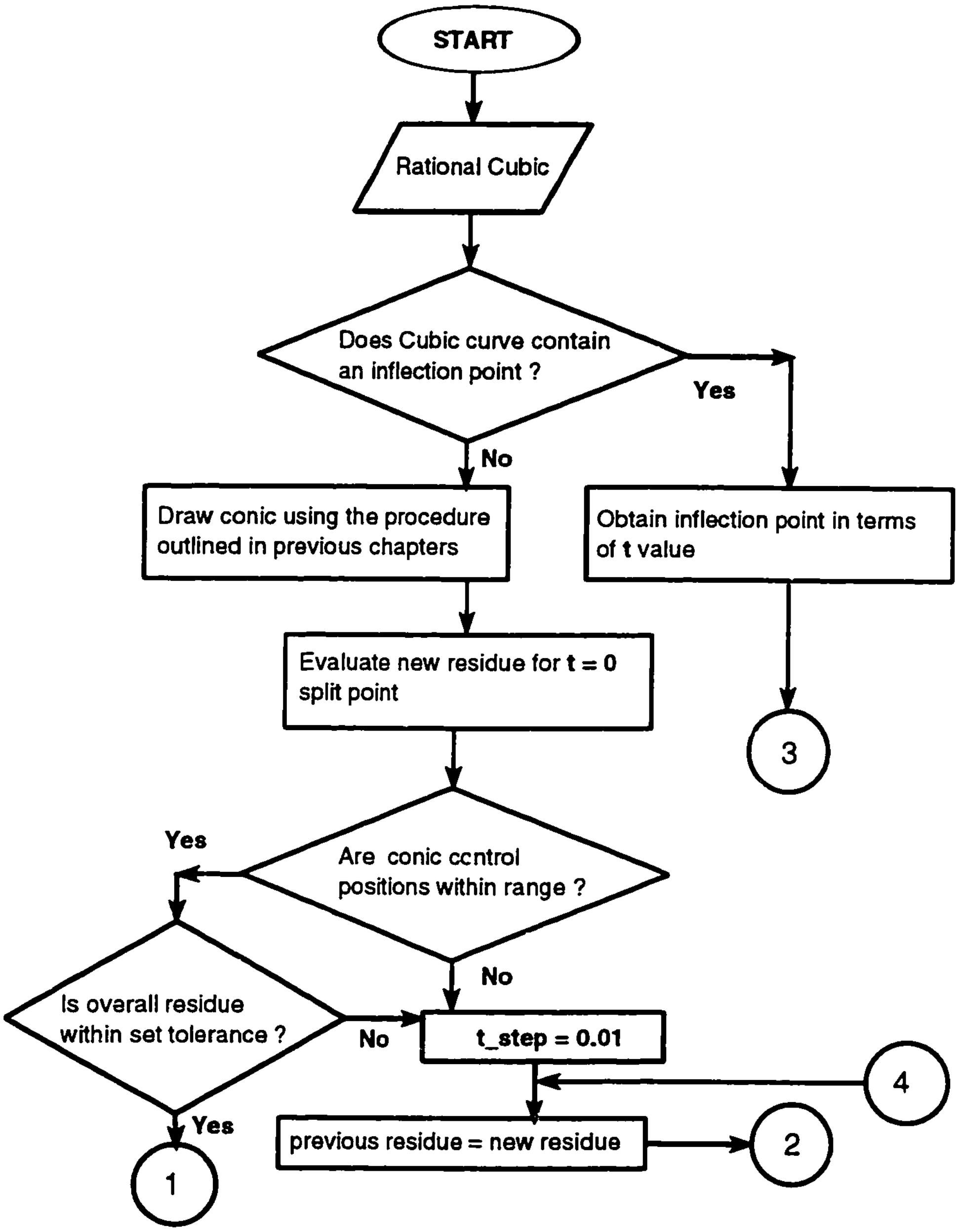


Figure 7.1: The conic rescue flow-chart using the method outlined in [30]as a future work.

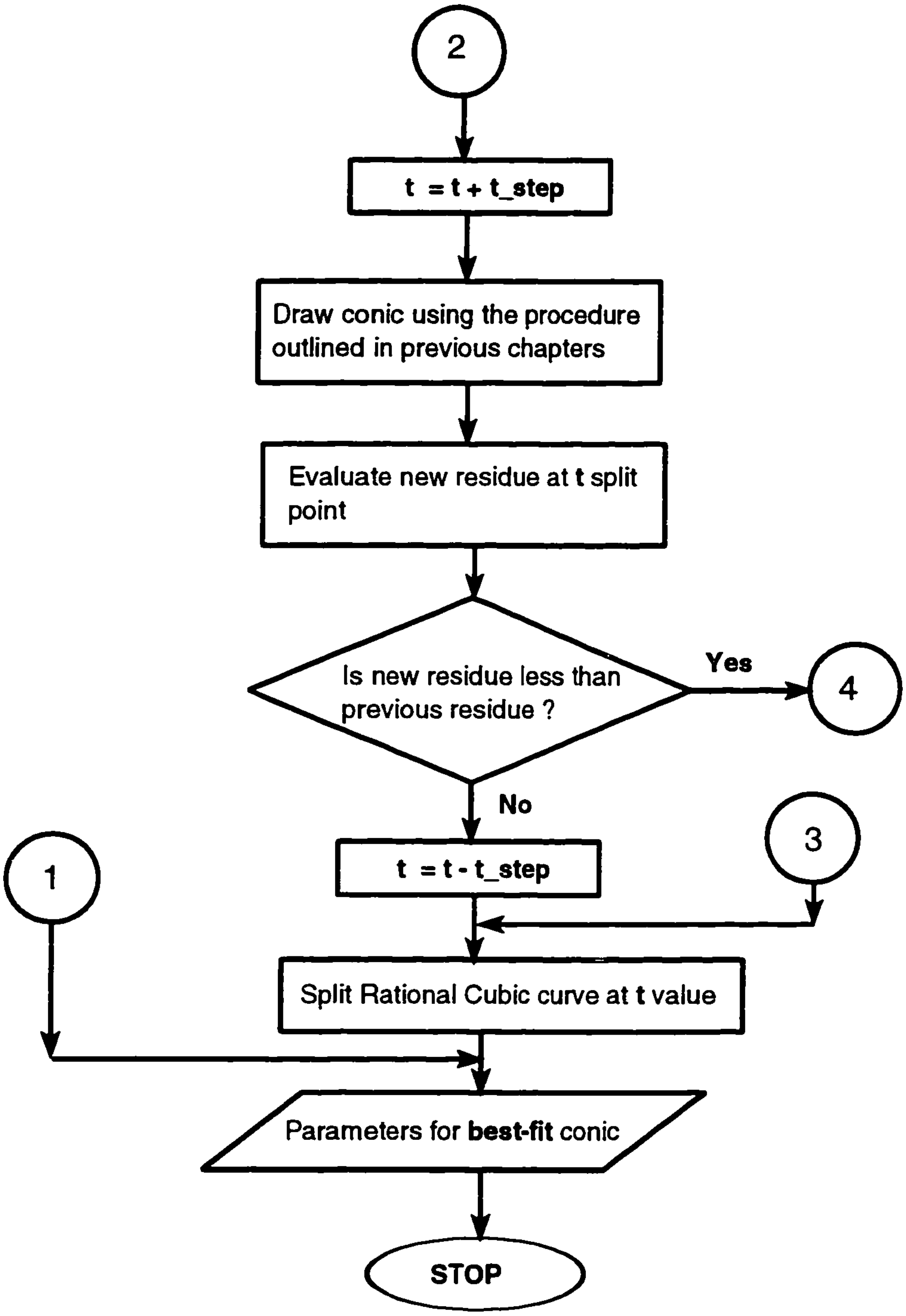


Figure 7.2: The conic rescue flow-chart using the method outlined in [30] as a future work.(*cont...*)

Bibliography

[1] G. Farin, "Algorithms for rational bezier curves", *Computer Aided Design*, vol. 15, no. 2, pp. 73–77, 1983.

[2] F. Hussain and M. L. V. Pitteway, "Quadratic representing of bezier cubic outlines", *Computer Science Dept.*, 1989, Brunel Univ., UXBRIDGE.

[3] V. Pratt, "Techniques for conic splines", *Proceedings of SIGGRAPH*, pp. 151–159, 1985.

[4] T. Pavlidis, "Curve fitting with conic splines", *ACM Transactions on Graphics*, pp. 1–31, 1983.

[5] D. Hearn and M. P. Baker, "Computer graphics", *Prentice Hall International*, 1994.

[6] D. Rogers and J. A. Adams, "Mathematical elements for computer graphics", *Mc-Graw Hill Publishing Company*, 1990.

[7] Foley F. Van Dam and Hughes, "Computer graphics : Principles and practise", *Addison-Wesley*, 1993, Second-edition.

[8] M. Plass and Maureen Stone, "Curve-fitting with Piecewise Parametric Cubics", *Computer Graphics*, vol. 17, no. 3, pp. 229–239, 1983.

[9] Laxmi Parida and S. P. Mudur, "Common Tangents to Planar Parametric Curves : a geometric solution", *Computer Aided Design*, vol. 27, no. 1, pp. 41–47, 1995.

[10] N. Sapidis and G. Farin, "Automatic fairing algorithm for B-Spline Curves", *Computer Aided Design*, vol. 22, no. 2, pp. 121–129, 1990.

[11] R. Loh, "Convex b-spline surfaces", *Computer Aided Design*, vol. 13, no. 3, pp. 145–149, 1981.

[12] W. Boehm, "Inserting new knots into B-Spline curves", *Computer Aided Design*, vol. 12, no. 4, pp. 199–201, 1980.

[13] M. Paluszny and R. Patterson, "A family of tangent continous cubic algebraic splines", *ACM Transactions on Graphics*, vol. 12, no. 3, pp. 209–232, 1993.

[14] J. C. Beatty R. Bartels and K. S. Booth, "Experimental comparision of splines using the shape-matching paradigm", *ACM Transactions on Graphics*, vol. 12, no. 3, pp. 179–208, 1993.

[15] B. A. Barsky, "Computer Graphics and Geometric Modeling using Beta-Splines", *Springer-verlag*, 1986, Tokyo.

[16] T. N. T. Goodman, "Properties of Beta-Splines ", *Journal of Approximation Theory*, vol. 44, no. 2, pp. 132–153, 1985.

[17] B. Barsky and J. Beatty, "Local control of bias and tension in Beta-Splines", *ACM Transactions on Graphics*, vol. 2, no. 2, pp. 73–77, 1983.

[18] B. Joe, "Discrete Beta-Splines", *Computer Graphics*, vol. 21, no. 4, pp. 137–144, 1987.

[19] D. Joe, "Multiple knot and rational cubic beta-splines", *ACM Transactions on Graphics*, vol. 8, no. 2, pp. 100–120, 1989.

[20] T. N. T. Goodman and K. Unsworth, "Manipulating shape and producing geometric continuity in beta-splines curves", *IEEE Computer Graphics and Applications*, vol. 6, no. 2, pp. 50–56, 1986.

[21] M. E. Hohmeyer and B. A. Barsky, "Rational continuity : Parametric, geometric, and frenet frame continuity of rational curves", *ACM Transactions on Graphics*, vol. 8, no. 4, pp. 335–359, 1989.

[22] G. Nielson, "Rectangular ν-splines", *IEEE Computer Graphics and Applications*, pp. 35–40, 1986.

[23] J. A. Gregory and M. Sarfraz, "A rational cubic spline with tension", *Computer Aided Geometric Design*, vol. 7, pp. 1–13, 1990.

[24] M. Sarfraz J. A. Gregory and P. K. Yuen, "Interactive curve design using c^2 rational splines", *Computers and Graphics*, vol. 18, no. 2, pp. 153–159, 1994.

[25] T. A. Foley and H. S. Ely, "Interpolation with interval and point tension controls using cubic weighted ν splines", *ACM Transactions on Mathematical Software*, vol. 13, no. 1, pp. 68–96, 1987.

[26] F. Hussain, "Conic rescue of bezier fonts", *Springer-verlag*, 1989, Tokyo.

[27] Azhar Sayeed, "Reducing the complexity of rational cubic for designing the objects", *Master Thesis*, 1996.

[28] G. Farin, "Curvature continuity and offsets for piecewise conics", *ACM Transactions on Graphics*, vol. 8, no. 2, pp. 89–99, 1989.

[29] L. Piegl, "Defining c^1 curves containing conic segments", *Computers and Graphics*, vol. 8, no. 2, pp. 177–182, 1984.

[30] H. Pottmann, "Locally controllable conic splines with curvature continuity", *ACM Transactions on Graphics*, vol. 10, no. 4, pp. 366–377, 1991.

[31] J. Sun K. Qin and X. Wang, "Representing conics using nurbs of degree two", *Computers and Graphics*, vol. 11, no. 5, pp. 285–291, 1992.

[32] Gerald Farin, "Surfaces for computer aided geometric design", *Academic Press*, 1988.

[33] Muhammad Sarfraz, "Designing of curves and surfaces using rational cubics", *Computers and Graphics*, vol. 17, no. 5, pp. 529–538, 1993.

[34] Muhammad Sarfraz, "A rational cubic spline with biased, point and interval tension", *Computers and Graphics*, vol. 16, pp. 427–430, 1992.

Vita

- Mohammed **Aiyaz** Hussain
- Born in 1973, at Hyderabad, INDIA.
- Received the Bachelor of Engineering degree in Computer Science and Engineering from Osmania University, Hyderabad, in August 1994.
- Joined the Information and Computer Sciences Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, as a Research Assistant in September 1995.
- Completed Masters Degree requirements in the Information and Computer Sciences Department in December 1997.
- Received several scholarships and merit certificates for excellence in studies.